



TUGAS AKHIR - TE 145561

**DESAIN DAN PROTOTIPE SISTEM DETEKSI GANGGUAN
IMPEDANSI TINGGI PADA SUTM 20 KV DENGAN SISTEM
PELAPORAN GANGGUAN MENGGUNAKAN KOMUNIKASI
DATA MODUL GSM**

Krisna Sadewa
NRP 2214038020
Iqbal Asad Kustsuro
2214038024

Dosen Pembimbing
Dr. Dimas Anton Asfani S.T., M.T.
Rachmad Setiawan S.T., M.T.

PROGRAM STUDI TEKNIK LISTRIK
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017

-----*Halaman ini sengaja dikosongkan*-----



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - TE 145561

***DESIGN AND PROTOTYPE HIGH IMPEDANCE FAULT
DETECTION SYSTEM AT 20 KV SUTM WITH FAULT
REPORTING SYSTEM USING GSM MODULE DATA
COMMUNICATION***

Krisna Sadewa
NRP 2214038020
Iqbal Asad Kutsuro
2214038024

Supervisor
Dr. Dimas Anton Asfani S.T., M.T.
Rachmad Setiawan S.T., M.T.

***ELECTRICAL ENGINEERING STUDY PROGRAM
Electrical and Automation Engineering Department
Vocational Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2017***

-----*Halaman ini sengaja dikosongkan*-----

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya dengan judul **“Desain dan Prototipe Sistem Deteksi Gangguan Impedansi Tinggi Pada SUTM 20 KV dengan Sistem Pelaporan Gangguan Menggunakan Komunikasi Data Modul GSM”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

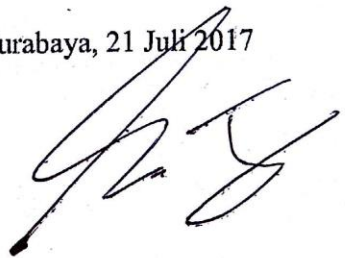
Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.



Krisna Sadewa
NRP 2214038020

Surabaya, 21 Juli 2017



Iqbal Asad Kutsuro
NRP 2214038024

-----*Halaman ini sengaja dikosongkan*-----

**DESAIN DAN PROTOTIPE SISTEM DETEKSI GANGGUAN
IMPEDANSI TINGGI PADA SUTM 20 KV DENGAN SISTEM
PELAPORAN GANGGUAN MENGGUNAKAN KOMUNIKASI
DATA MODUL GSM**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya
Pada**

**Program Studi Teknik Listrik
Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember**

Menyetujui:

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Dimas Anton Asfari S.T., M.T.

Rachmad Setiawan S.T., M.T.

NIP. 19810905 200501 1002

NIP. 19690529 199512 1001

-----*Halaman ini sengaja dikosongkan*-----

DESAIN DAN PROTOTYPE SISTEM DETEKSI GANGGUAN IMPEDANSI TINGGI PADA SUTM 20 KV DENGAN SISTEM PELAPORAN GANGGUAN MENGGUNAKAN KOMUNIKASI DATA MODUL GSM

Nama Mahasiswa : Krisna Sadewa
NRP : 2214038020
Nama Mahasiswa : Iqbal Asad Kutsuro
NRP : 2214038024
Pembimbing I : Dr. Dimas Anton Asfani. ST., MT.
NIP : 19810905 200501 1 002
Pembimbing II : Rachmad Setiawan. ST., MT.
NIP : 19690529 199512 1 001

ABSTRAK

Berdasarkan data yang telah diamati dilapangan, setiap jaringan dan sistem distribusi listrik selalu mengalami gangguan hubung singkat. Gangguan ini dapat terjadi akibat banyak faktor salah satunya yakni faktor eksternal atau faktor alam yang dapat berupa ranting pohon atau karena hewan. Gangguan hubung singkat yang biasanya terjadi mempunyai impedansi tinggi sehingga sangat sulit dideteksi oleh alat yang ada sekarang akibat arus gangguan yang relatif kecil terutama saat lokasi gangguan terletak di titik paling jauh dari sumber. Selain itu juga mempengaruhi tegangan kerja di sistem distribusi 20 Kv dan adanya gangguan menyebabkan *drop* tegangan yang menyebabkan PLN tidak dapat memenuhi beban di daerah yang terkena gangguan.

Untuk merealisasikan alat ini diperlukan simulasi jaringan tegangan menengah yang dapat menghasilkan gangguan hubung singkat satu fasa ke tanah dan antar fasa, mikrokontroler akan menangkap besar arus gangguan yang terjadi dan mengirimkan laporan lokasi gangguan melalui media SMS.

Tugas akhir ini dapat memberikan laporan berupa besar arus, jarak lokasi gangguan, fasa yang mengalami gangguan dan waktu terjadinya gangguan. Pengiriman laporan sms gangguan rata-rata mempunyai waktu pengiriman 8,7 detik. Kondisi ini masih bergantung pada kondisi sinyal dari *provider* yang digunakan.

Kata Kunci : Gangguan satu fasa ke tanah, gangguan antar fasa, Impedansi tinggi, Sistem distribusi 20 kV.

-----*Halaman ini sengaja dikosongkan*-----

**DESIGN AND PROTOTYPE HIGH IMPEDANCE FAULT
DETECTION SYSTEM AT 20 KV SUTM WITH FAULT
REPORTING SYSTEM USING GSM MODULE DATA
COMMUNICATION**

Name	: Krisna Sadewa
Registration Number	: 2214038020
Name	: Iqbal Asad Kutsuro
Registration Number	: 2214038024
Supervisor I	: Dr. Dimas Anton Asfani. ST., MT.
ID	: 19810905 200501 1 002
Supervisor II	: Rachmad Setiawan. ST., MT.
ID	: 19690529 199512 1 001

ABSTRACT

Based on the data that has been observed in the field, each network and electrical distribution system always experience Short Circuit interruption. This short-circuit disruption can occur due to many factors one of which is the external factor or natural factor that can be a branch of a tree or animal. The usual Short Circuit impedance has a high impedance, which is very difficult to detect by the current equipment due to relatively small fault current especially when the disturbance location is located at the farthest point from the source. In addition, the Short Circuit also affects the working stress in 20 kV distribution system. Due to short-circuit impedance high impedance causes a Voltage drop that causes PLN can not meet the load in the affected area.

To realize this tool is required simulation medium Voltage network that can generate Short Circuit one phase to ground and phase, microcontroller will catch big current of disturbance that happened and sending report of location of disturbance through SMS media.

This final project can give report in the form of big of current, distance of disturbance location, phase of disturbance and time of disturbance. Submission of sms reports Average interruption has a delivery time of 8.7 seconds. This condition is still dependent on the signal condition of the provider used..

Keywords: *One-phase to ground disturbance, inter-phase fault, High impedance, 20 kV distribution system.*

-----*Halaman ini sengaja dikosongkan*-----

KATA PENGANTAR

Alhamdulillah kami panjatkan kepada Allah Subhanahu Wa Ta'ala, atas limpahan rahmat dan kemudahan dariNya, hingga kami dapat menyelesaikan tugas akhir ini dengan baik, begitu pula dengan pembuatan buku tugas akhir ini.

Tugas akhir ini dilakukan untuk memenuhi beban satuan kredit semester (SKS) yang harus ditempuh sebagai persyaratan akademis di Jurusan D3 Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya untuk menyelesaikan program pendidikan Diploma di Teknik Elektro dengan judul :

Desain dan Prototipe Sistem Deteksi Gangguan Impedansi Tinggi pada SUTM 20 kV dengan Sistem Pelaporan Gangguan Menggunakan Komunikasi Data Modul GSM

Penulis mengucapkan terima kasih kepada Ibu dan Bapak penulis yang memberikan berbagai bentuk doa serta dukungan tulus tiada henti, serta Bapak Dr. Dimas Anton Asfani S.T., M.T. dan Bapak Rachmad Setiawan S.T., M.T. atas segala bimbingan ilmu, moral, dan spiritual dari awal hingga terselesaikannya Tugas Akhir ini. Penulis juga mengucapkan banyak terima kasih kepada semua pihak yang telah membantu baik secara langsung maupun tidak langsung dalam proses penyelesaian Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat dalam pengembangan keilmuan di kemudian hari.

Surabaya 21 Juli 2017

Penulis

-----*Halaman ini sengaja dikosongkan*-----

DAFTAR ISI

	HALAMAN
HALAMAN JUDUL	i
PERNYATAAN KEASLIAN TUGAS AKHIR	v
ABSTRAK.....	ix
<i>ABSTRACT</i>	xi
KATA PENGANTAR	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR.....	xix
DAFTAR TABEL	xxiii
 BAB I PENDAHULUAN	 1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Metodologi Penelitian	3
1.6 Sistematika Laporan.....	4
1.7 Relevansi.....	4
 BAB II TEORI DASAR	 5
2.1 Tinjauan Pustaka	5
2.2 Jaringan Distribusi Tegangan Menengah	6
2.2.1 Jenis Jaringan Distribusi Primer	7
2.3 Gangguan Hubung Singkat	8
2.3.1 Penyebab Gangguan Hubung Singkat	9
2.3.1 Jenis Gangguan Hubung Singkat	9
2.4 <i>Transformator Step Down</i>	12
2.5 NGR (<i>Neutral Grounding Resistance</i>)	13
2.6 Perhitungan Arus Hubung Singkat	15
2.6.1 Menghitung Impedansi Sumber	15
2.6.2 Menghitung Reaktansi Trafo	16
2.6.3 Menghitung Impedansi Penyulang	17
2.6.4 Menghitung Impedansi Ekuivalen Jaringan	17
2.6.5 Menghitung Arus Gangguan Antar Fasa	19
2.6.6 Menghitung Arus Gangguan Satu Fasa ke Tanah	20
2.7 Sensor Arus YHDC ST-013	21
2.8 Sensor Tegangan	22
2.9 <i>Board Arduino Mega</i>	23

2.10 Arduino IDE	24
2.11 LCD (<i>Liquid Crystal Display</i>)	25
2.12 <i>Real Time Clock</i> (RTC)	26
2.13 SIM 900 atau GPRS dan <i>GSM Shield</i>	27
2.14 Modem <i>Maestro</i> 100 GSM/GPRS Serial	28
2.15 <i>MIT App Inventor</i>	29
2.16 <i>Delphi</i>	30
BAB III PERANCANGAN DAN PEMBUATAN ALAT	33
3.1 Perancangan Sistem Keseluruhan	33
3.2 Perancangan Mekanik.....	35
3.2.1 Perancangan Modul Gangguan.....	36
3.2.1.1 <i>Trafo Step Down</i> 380V / 12V.....	37
3.2.1.2 Penentuan Besaran Tahanan.....	38
3.2.1.3 Penentuan Nilai NGR (<i>Neutral Grounding Resistance</i>).....	40
3.2.2 Perancangan Panel <i>Box</i>	40
3.3 Perancangan <i>Hardware</i>	41
3.3.1 LCD (<i>Liquid Crystal Display</i>).....	42
3.3.2 Perancangan <i>Power Supply</i>	43
3.3.3 RTC (<i>Real Time Clock</i>).....	44
3.3.4 Perancangan SIM900	45
3.3.5 Sensor Arus	46
3.3.6 Sensor Tegangan	47
3.4 Perancangan <i>Software</i>	49
3.4.1 Pemrograman Sistem Arduino IDE.....	49
3.4.2 Pemrograman pada <i>App Inventor</i>	53
3.4.3 Pemrograman pada <i>Delphi</i>	60
BAB IV PENGUJIAN DAN ANALISA DATA.....	63
4.1 Pengujian Arduino Mega 2560	64
4.2 Pengujian <i>Power Supply</i>	67
4.3 Pengujian <i>Real Time Clock</i> (RTC)	68
4.4 Pengujian <i>Liquid Crystal Display</i> (LCD)	70
4.5 Pengujian CT Sensor YHDC ST-013	72
4.6 Pengujian Sensor Tegangan.....	78
4.7 Pengujian Komunikasi SIM900.....	85
4.7.1 Pengujian SIM900.....	81
4.7.2 Pengujian Modem <i>Maestro</i>	84
4.8 Pengujian <i>Short Circuit</i> dan <i>Open Circuit Transformator</i>	90

4.8.1 Perhitungan Impedansi Dalam Trafo	91
4.8.2 Perhitungan Impedansi Ekvivalen Trafo	92
4.8.3 Perhitungan <i>Per Unit</i> (PU)	93
4.9 Pengujian Arus Hubung Singkat	93
4.10 Pengujian Modul Gangguan.....	95
4.11 Pengujian Keseluruhan.....	99
4.11.1 Gangguan Fasa S.....	101
4.11.2 Gangguan Fasa R.....	104
4.11.3 Gangguan Fasa T.....	107
4.11.4 Gangguan Fasa S – T.....	110
4.11.5 Gangguan Fasa R – S.....	113
4.12 <i>Database</i> Aplikasi Android dan <i>Delphi</i>	116
4.13 Analisa Relevansi.....	117
BAB V PENUTUP.....	119
5.1 Kesimpulan	119
5.2 Saran	120
DAFTAR PUSTAKA	121
LAMPIRAN A.....	123
A.1. <i>Listing</i> Program pada Arduino.....	123
A.2. <i>Listing</i> Block Editor <i>MIT App Inventor</i>	143
A.3. <i>Listing</i> Program pada <i>Delphi 7</i>	147
LAMPIRAN B	153
B.1. <i>DATASHEET</i> RTC DS3231	153
B.2. <i>DATASHEET</i> SIM900.....	160
B.3. <i>DATASHEET</i> Arduino Mega	167
B.4. <i>DATASHEET</i> Sensor Arus YHDC SCT-013	168
DAFTAR RIWAYAT HIDUP.....	169
DAFTAR RIWAYAT HIDUP.....	171

-----*Halaman ini sengaja dikosongkan*-----

DAFTAR GAMBAR

HALAMAN

Gambar 2.1	Sistematika Jaringan Distribusi Tegangan Menengah .	6
Gambar 2.2	Sistem Jaringan <i>Radial</i>	7
Gambar 2.3	Sistem Jaringan <i>Loop</i>	8
Gambar 2.4	Gangguan Hubung Singkat Tiga Fasa.....	10
Gambar 2.5	Gangguan Hubung Singkat Tiga Fasa ke Tanah	10
Gambar 2.6	Gangguan Hubung Singkat Dua Fasa	11
Gambar 2.7	Gangguan Hubung Singkat Dua Fasa ke Tanah.....	11
Gambar 2.8	Gangguan Satu Fasa ke Tanah	11
Gambar 2.9	Konfigurasi <i>Wye-Delta</i>	12
Gambar 2.10	Konfigurasi <i>Delta-Wye</i>	13
Gambar 2.11	<i>Neutral Grounding Resistance</i>	14
Gambar 2.12	<i>Neutral Grounding Liquid</i>	14
Gambar 2.13	<i>Neutral Grounding Resistance Solid</i>	15
Gambar 2.14	Skema Perancangan Arus Hubung Singkat.....	15
Gambar 2.15	Sensor Arus YHDC ST-013.....	22
Gambar 2.16	Rangkaian Penyearah Gelombang Penuh	23
Gambar 2.17	<i>Board</i> Arduino Mega	24
Gambar 2.18	Jendela Arduino IDE.....	24
Gambar 2.19	LCD (<i>Liquid Crystal Display</i>)	26
Gambar 2.20	Modul RTC	27
Gambar 2.21	Skematik RTC.....	27
Gambar 2.22	Modul SIM 900.....	28
Gambar 2.23	Modem <i>Maestro</i> 100.....	29
Gambar 2.24	Antarmuka MIT <i>App Inventor</i>	30
Gambar 2.25	Jendela <i>Design Delphi</i>	31
Gambar 3.1	Penyulang Mayjend Sungkono	34
Gambar 3.2	Diagram Fungsional Sistem	35
Gambar 3.3	Skema Rangkaian Modul Gangguan.....	36
Gambar 3.4	Skema Prototipe Modul Gangguan	37
Gambar 3.5	Trafo <i>Step Down</i> 380/12 Volt.....	38
Gambar 3.6	Panel <i>Box</i>	41
Gambar 3.7	Isi Panel <i>Box</i>	42
Gambar 3.8	Rangkaian LCD	43
Gambar 3.9	Skema <i>Powerbank</i> ke Arduino Mega	44

Gambar 3.10	Skema Perancangan RTC	45
Gambar 3.11	Skema <i>Shield</i> SIM900 dengan Arduino Mega 2560...45	45
Gambar 3.12	Sensor Arus YHDC	46
Gambar 3.13	Rangkaian Pengkondisi Sinyal	47
Gambar 3.14	Skema Rangkaian <i>Rectifier</i>	48
Gambar 3.15	Skema Rangkaian Pembagi Tegangan.....	48
Gambar 3.16	<i>Flowchart</i> 1 Program Arduino.....	51
Gambar 3.17	<i>Flowchart</i> 2 Program Arduino.....	52
Gambar 3.18	<i>Flowchart</i> 3 Program Arduino.....	53
Gambar 3.19	Tampilan <i>Screen</i> 1	54
Gambar 3.20	<i>Flowchart Screen</i> 1.....	55
Gambar 3.21	Tampilan <i>Screen</i> 2	56
Gambar 3.22	<i>Flowchart Screen</i> 2.....	57
Gambar 3.23	Tampilan <i>Screen</i> 3	58
Gambar 3.24	<i>Flowchart Screen</i> 3.....	59
Gambar 3.25	Tampilan HMI (<i>Human Machine Interface</i>) pada <i>Delphi</i>	60
Gambar 3.26	<i>Flowchart</i> Pemrograman HMI pada <i>Delphi</i>	61
Gambar 4.1	Pengujian Keseluruhan Sistem	63
Gambar 4.2	<i>Flowchart</i> Pengujian Arduino	65
Gambar 4.3	Pengujian Arduino	65
Gambar 4.4	Skema Pengujian Arus <i>Power Supply</i>	67
Gambar 4.5	Skema Pengujian <i>Power Supply</i>	68
Gambar 4.6	<i>Flowchart</i> Pengujian RTC.....	69
Gambar 4.7	Pengujian RTC	70
Gambar 4.8	<i>Flowchart</i> Pengujian LCD.....	71
Gambar 4.9	Hasil Tampilan LCD	72
Gambar 4.10	Skema Pengujian CT Sensor	73
Gambar 4.11	Program Pengujian CT Sensor.....	74
Gambar 4.12	Linierisasi CT Sensor 1	75
Gambar 4.13	Linierisasi CT Sensor 2	76
Gambar 4.14	Linierisasi CT Sensor 3	76
Gambar 4.15	Program Tampilan LCD Pengujian CT Sensor	77
Gambar 4.16	Skema Pengujian Sensor Tegangan.....	79
Gambar 4.17	Program Pengujian Sensor Tegangan	79
Gambar 4.18	Linierisasi Sensor Tegangan ADC 1	81
Gambar 4.19	Linierisasi Sensor Tegangan ADC 2	81
Gambar 4.20	Linierisasi Sensor Tegangan ADC 3	82
Gambar 4.21	Program Tampilan LCD Pengujian Sensor Tegangan	83

Gambar 4.22	Kalibrasi Sensor Tegangan	84
Gambar 4.23	<i>Flowchart</i> Pengujian SMS SIM900	85
Gambar 4.24	Hasil Pengujian SIM900	86
Gambar 4.25	Hasil Pengujian Waktu SMS	87
Gambar 4.26	Pengujian Modem <i>Maestro</i>	88
Gambar 4.27	Hasil Pengujian Modem <i>Maestro</i>	89
Gambar 4.28	Skema Pengujian <i>Short Circuit</i> Trafo.....	91
Gambar 4.29	Skema Pengujian Arus Hubung Singkat	94
Gambar 4.30	Skema Pengujian Rangkaian.....	95
Gambar 4.31	Hubung Singkat 1 KM	96
Gambar 4.32	Hubung Singkat 2 KM	97
Gambar 4.33	Hubung Singkat 4 KM	97
Gambar 4.34	Hubung Singkat Fasa S dan T	98
Gambar 4.35	Hubung Singkat Fasa R dan S.....	99
Gambar 4.36	Modul Gangguan pada Kondisi Normal	100
Gambar 4.37	Tampilan LCD saat Keadaan Normal	100
Gambar 4.38	Modul Gangguan saat Gangguan Fasa S	101
Gambar 4.39	Panel saat Gangguan Fasa S	102
Gambar 4.40	Gangguan Fasa S pada Aplikasi Android	102
Gambar 4.41	Gangguan Fasa S pada Aplikasi <i>Server</i>	103
Gambar 4.42	Modul Gangguan saat Gangguan Fasa R.....	104
Gambar 4.43	Panel Gangguan Fasa R	105
Gambar 4.44	Gangguan Fasa R pada Aplikasi Android	105
Gambar 4.45	Gangguan Fasa R pada Aplikasi <i>Server</i>	106
Gambar 4.46	Modul Gangguan saat Gangguan Fasa T	107
Gambar 4.47	Panel Gangguan Fasa T	108
Gambar 4.48	Gangguan Fasa T pada Aplikasi Android	108
Gambar 4.49	Gangguan Fasa T pada Aplikasi <i>Server</i>	109
Gambar 4.50	Modul Gangguan saat Gangguan Fasa S-T.....	110
Gambar 4.51	Panel saat Gangguan S – T	111
Gambar 4.52	Gangguan Fasa S-T pada Aplikasi Android.....	111
Gambar 4.53	Gangguan S-T pada Aplikasi <i>Server</i>	112
Gambar 4.54	Modul Gangguan saat Gangguan Fasa R-S	113
Gambar 4.55	Panel Gangguan saat Gangguan R – S	114
Gambar 4.56	Gangguan R-S pada Aplikasi Android.....	114
Gambar 4.57	Gangguan R-S pada Aplikasi <i>Server</i>	115
Gambar 4.58	<i>Database</i> SMS pada Aplikasi Android	116
Gambar 4.59	<i>Database</i> SMS pada Aplikasi <i>Delphi</i>	117

-----*Halaman ini sengaja dikosongkan*-----

DAFTAR TABEL

	HALAMAN
Tabel 2.1 Perhitungan Impedansi Urutan Positif dan Negatif	17
Tabel 2.2 Perhitungan Impedansi Urutan Nol	17
Tabel 2.3 Perhitungan Ekivalen Positif dan Negatif	18
Tabel 2.4 Perhitungan Ekivalen Urutan Nol	19
Tabel 2.5 Perhitungan Arus Hubung Singkat 2 Fasa	20
Tabel 2.6 Perhitungan Arus Hubung Singkat Fasa ke Tanah	21
Tabel 2.7 Spesifikasi Sensor Arus YHDC-013	22
Tabel 2.8 Spesifikasi SIM 900	28
Tabel 2.9 Spesifikasi Modem <i>Maestro</i> 100	29
Tabel 3.1 Tahanan dan Reaktansi Kawat Penghantar	39
Tabel 4.1 Hasil Pengujian Arduino Mega 2560	66
Tabel 4.2 Pengujian <i>Powerbank</i>	68
Tabel 4.3 Hasil Pengujian RTC	70
Tabel 4.4 Hasil Sensing Awal CT Sensor	74
Tabel 4.5 Hasil Pengujian CT Sensor	78
Tabel 4.6 Data Linierisasi Sensor Tegangan	79
Tabel 4.7 Data Pengujian Sensor Tegangan	83
Tabel 4.8 Hasil Pengujian SIM900	87
Tabel 4.9 Hasil Pengujian Modem <i>Maestro</i>	89
Tabel 4.10 Hasil Pengujian <i>Transformator</i>	91
Tabel 4.11 Data Impedansi Dalam Trafo	92
Tabel 4.12 Data Impedansi Ekivalen Trafo	92
Tabel 4.13 Data Per Unit Trafo	93
Tabel 4.14 Data Arus Hubung Singkat Satu Fasa ke Tanah	94
Tabel 4.15 Data Arus Hubung Singkat Antar Fasa	94
Tabel 4.16 Data Pengujian Saluran Modul Gangguan	95
Tabel 4.17 Data Pengujian Beban Modul Gangguan	95
Tabel 4.18 Data Pengujian <i>Resistor</i> Modul Gangguan	96
Tabel 4.19 Data Pengujian Hubung Singkat 1 KM	96
Tabel 4.20 Data Pengujian Hubung Singkat 2 KM	97
Tabel 4.21 Data Pengujian Hubung Singkat 4 KM	97
Tabel 4.22 Data Pengujian Hubung Singkat Fasa S dan T	98
Tabel 4.23 Data Pengujian Hubung Singkat Fasa R dan S	99
Tabel 4.24 Hasil Pengujian Gangguan Fasa S	103
Tabel 4.25 Hasil Pengujian Gangguan Fasa R	106
Tabel 4.26 Hasil Pengujian Gangguan Fasa T	109

Tabel 4.27 Hasil Pengujian Gangguan Fasa S - T.....	112
Tabel 4.28 Hasil Pengujian Gangguan Fasa R - S	115

BAB I

PENDAHULUAN

1.1 Latar Belakang

Berdasarkan data, 70% sampai dengan 90% dari seluruh gangguan yang mengenai SUTM PLN di Indonesia adalah gangguan yang bersifat temporer (sementara). Gangguan hubung singkat fasa ke tanah adalah gangguan yang bersifat temporer (sementara) yang paling sering terjadi dikarenakan menempelnya hewan atau ranting pohon yang basah. Ketika terjadi gangguan fasa ke tanah petugas perlu mencari gangguan dengan inspeksi secara *visual* atau mengukur tahanan penghantar di setiap *section*. Cara ini kurang efektif bila gangguan terjadi di penyulang yang panjang dan memiliki banyak *section*. Oleh karena itu perlu adanya sistem monitoring gangguan ke tanah dan deteksi lokasi gangguan menggunakan sensor arus dan tegangan yang diletakkan pada saluran fasa dan pentanahan di sisi pelanggan sebagai pembaca arus saat terjadinya gangguan ke tanah. Keluaran sensor arus tersebut diterjemahkan oleh ADC ke mikrokontroler (Arduino).

Untuk merealisasikan alat ini diperlukan simulasi jaringan tegangan menengah yang dapat menghasilkan gangguan fasa ke tanah dan fasa ke fasa, mikrokontroler nantinya akan menangkap adanya indikasi gagguan yang terjadi lalu mengirimkan laporan berupa data yang berisi besar arus gangguan, jenis gangguan, fasa yang terjadi gangguan, waktu terjadi gangguan, dan lokasi gangguan ke aplikasi pada *smartphone* dan ke aplikasi *server* pada komputer yang dibuat menggunakan *software Delphi* sehingga memudahkan pendataan gangguan.

Tugas akhir ini dapat memberikan laporan berupa besar arus, jarak lokasi gangguan, fasa yang mengalami gangguan dan waktu terjadinya gangguan dengan cara merubah arus yang terjadi tersebut kemudian akan diterima sebagai data oleh ADC (*Analog to Digital Converter*) yang akan menghasilkan data digital yang yang akan ditampilkan dalam besaran kilometer oleh mikrokontroler. Alat ini menggunakan beberapa *resistor* yang diseri sebagai kabel fasa dan sakelar untuk menghasilkan gangguan. Kemudian Arduino akan menampilkan informasi pada *display* HMI.

Diharapkan alat ini nantinya akan membantu mempercepat proses perbaikan jaringan dan akan menambah keandalan sistem yang dimiliki PLN.

1.2 Permasalahan

Beberapa permasalahan yang menjadi objek penulisan tugas akhir ini adalah:

1. Dalam penyaluran tenaga listrik jaringan tegangan menengah sering kali terjadi gangguan fasa ke tanah yang menyebabkan bekerjanya relai dan jatuhnya pemutus tenaga. Gangguan ini sangat merugikan pelanggan-pelanggan listrik.
2. Belum terdapat sistem yang mampu untuk mencatat fasa mana yang mengalami gangguan ketika terjadi gangguan satu fasa ke tanah pada sistem jaringan tegangan menengah.
3. Ketika terjadi gangguan pada tegangan menengah, pihak PLN juga perlu turun kelapangan untuk mencari lokasi gangguan yang terjadi secara manual dengan menghitung tahanan dengan *Megger*.
4. Dalam penanganan gangguan masih ada ketergantungan terhadap pelaporan masyarakat.
5. Pendataan pada *server* masih menggunakan cara manual yang sangat rumit.

1.3 Batasan Masalah

Dalam pembuatan tugas akhir ini, diberikan batasan permasalahan agar pembahasan tidak meluas dan menyimpang dari tujuan. Pada tugas akhir ini menggunakan Arduino Mega dan media komunikasi menggunakan modem GSM pada SIM900 dan batasan masalah lainnya sebagai berikut :

1. Ruang lingkup penggunaan alat ini hanya pada sisi pelanggan tegangan menengah dengan tegangan 20 kV.
2. Jenis gangguan yang akan dideteksi oleh alat ini hanya gangguan satu fasa ke tanah dan fasa ke fasa.
3. Media komunikasi alat ini menggunakan SMS yang nantinya akan ditampilkan pada aplikasi *handphone* Android sebagai media antar muka (*interface*) dan ke *server* di komputer.
4. Alat dapat digunakan pada penyulang khusus dimana tidak ada percabangan pada jaringan SUTM.
5. Sensor arus dan tegangan yang digunakan disesuaikan dengan kemampuan mikrokontroler yang digunakan.

1.4 Tujuan

Tujuan Krisna Sadewa menuliskan tugas akhir ini adalah :

1. Mendeteksi gangguan impedansi tinggi (satu fasa ke tanah dan antar fasa) yang terjadi di jaringan SUTM 20 kV
2. Mempercepat jalannya informasi mengenai deteksi gangguan kepada komputer *server* dan *user*.

Tujuan Iqbal Asad Kutsuro menuliskan tugas akhir ini adalah :

1. Membuat rancang bangun gangguan impedansi tinggi (satu fasa ke tanah dan antar fasa) yang terjadi di jaringan SUTM 20 kV.
2. Mensimulasikan gangguan impedansi tinggi yang terjadi di jaringan SUTM 20 kV

1.5 Metodologi Penelitian

Dalam pelaksanaan tugas akhir prototipe sistem pelaporan gangguan jaringan tegangan menengah berbasis Arduino dengan media SMS, ada beberapa kegiatan yang dapat diuraikan. Tahap pertama adalah tahap persiapan, pada tahap ini akan dilakukan studi literature mengenai jenis gangguan yang terjadi pada sistem JTM, metode pendeteksian lokasi gangguan pada JTM, Mempelajari karakteristik kerja sensor arus dan sensor tegangan. Kemudian yang terakhir adalah mempelajari sistem komunikasi antara *hardware* (Arduino) dengan *handphone* sebagai media *display* dengan memanfaatkan modul GSM. Tahap kedua adalah tahap perancangan, pada tahap ini akan dilakukan perancangan sesuai data yang telah didapatkan dari studi literatur. Berikutnya adalah tahap pembelian dan pembuatan, Pada tahap pembelian komponen ini akan dilakukan pembelian komponen sesuai data yang telah dikumpulkan melalui studi literatur. Tahap pembuatan pada tahap ini akan dilakukan pembuatan alat setelah semua komponen telah lengkap disertai dengan data cara pembuatannya yang diperoleh dari studi literatur. Tahap selanjutnya adalah tahap pengujian, Pada tahap ini akan dilakukan pengujian alat yang telah dibuat. Tahap pengujian ini meliputi pengujian *hardware* maupun *software*. Tahap analisa pada tahap ini akan dilakukan analisa. Faktor apa saja yang menyebabkan alat tidak bekerja sesuai dengan keinginan atau terjadi *error*. Tahap akhir pada tahap ini akan dilakukan penyempurnaan pada alat dan membenahi alat jika terjadi *error* sesuai dengan data yang telah didapat pada analisa.

1.6 Sistematika Laporan

Pembahasan tugas akhir ini akan dibagi menjadi lima Bab dengan sistematika sebagai berikut:

Bab I Pendahuluan

Bab ini meliputi latar belakang, permasalahan, tujuan penelitian, metodologi penelitian, sistematika laporan, dan relevansi.

Bab II Teori Dasar

Bab ini menjelaskan tentang tinjauan pustaka, konsep dari metode yang digunakan dalam mencari lokasi gangguan pada jaringan tegangan menengah.

Bab III Perancangan Sistem Kontrol

Bab ini membahas desain dan perancangan *software* maupun *hardware* dari alat deteksi lokasi gangguan satu fasa ke tanah dan fasa ke fasa.

Bab IV Pengujian dan Analisa Data

Bab ini memuat hasil simulasi dan implementasi serta analisa dari hasil tersebut.

Bab V Penutup

Bab ini berisi kesimpulan dan saran dari hasil pembahasan yang telah diperoleh.

1.7 Relevansi

Hasil yang diperoleh dari tugas akhir ini diharapkan agar mampu mempermudah petugas PLN dalam menentukan letak gangguan impedansi tinggi yang terjadi pada jaringan distribusi. Selain itu juga diharapkan mampu untuk mempercepat kinerja PLN dalam penanganan gangguan jaringan tegangan menengah.

BAB II

TEORI DASAR

Pada bab ini menjelaskan mengenai teori-teori yang mendasari perancangan dan pembuatan tugas akhir yang dibuat oleh Krisna Sadewa dan Iqbal Asad Kutsuro. Teori yang mendukung Krisna Sadewa dan Iqbal Asad Kutsuro dalam penyelesaian tugas akhir ini diantaranya adalah mengenai arus gangguan hubung singkat impedansi tinggi dan komunikasi pelaporan gangguan.

Untuk Iqbal Asad Kutsuro teori penunjang yang dibutuhkan diantaranya yakni jenis arus gangguan hubung singkat pada Sub Bab 2.3.1, sistem jaringan tenaga listrik pada Sub Bab 2.2, trafo *Step Down* pada Sub Bab 2.4, sensor arus pada Sub Bab 2.7, dan sensor tegangan pada Sub Bab 2.8.

Sedangkan Untuk Krisna Sadewa teori penunjang yang dibutuhkan diantaranya yakni mikrokontroler pada Sub Bab 2.9, RTC pada Sub Bab 2.12, LCD pada Sub Bab 2.11, modem GSM SIM900 pada Sub Bab 2.13, Modem *Maestro* 100 pada Sub Bab 2.14, *Delphi* pada Sub Bab 2.16 dan *App Inventor* pada Sub Bab 2.15.

2.1 Tinjauan Pustaka

Penyelesaian masalah untuk mengetahui titik gangguan pada jaringan tegangan menengah saat ini di lapangan masih menggunakan cara atau metode yang lama. Metode ini menggunakan *Sectionalizer* atau LBS. Dengan menggunakan *Sectionalizer*, satu penyulang terbagi menjadi beberapa *section*. Sehingga jika terjadi gangguan maka satu penyulang akan mati dan pelokalisasi gangguan dengan membuka semua *Sectionalizer* sepanjang penyulang. Setelah itu, petugas masih melakukan pengukuran besaran tahanan dengan menggunakan *Megger* setiap *section* yang dilepas untuk mengetahui pada *section* manakah gangguan terjadi.

Untuk mengetahui titik gangguan caranya adalah dengan memonitoring besar arus gangguan hubung singkat menggunakan mikrokontroler. Dengan metode yang digunakan hanya dapat menentukan *section* mana yang terkena gangguan dalam satu penyulang untuk pelokalisasi titik gangguan dan metode tersebut mempercepat pelokalisasi gangguan tanpa membuka semua *Sectionalizer* dalam satu penyulang.

Pada tugas akhir ini akan dilakukan perancangan prototipe sistem pelaporan gangguan beserta posisi gangguan. Simulasi gangguan yang digunakan pada dan penentuan jarak gangguan hubung singkat dihitung dengan menggunakan hukum *Ohm*. Hasil yang diharapkan metode ini adalah dapat menentukan lokasi titik gangguan berapa kilometer dari gardu induk dan dapat mempercepat pendeteksian gangguan dan lokasi gangguan pada jaringan tegangan menengah 20 kV.

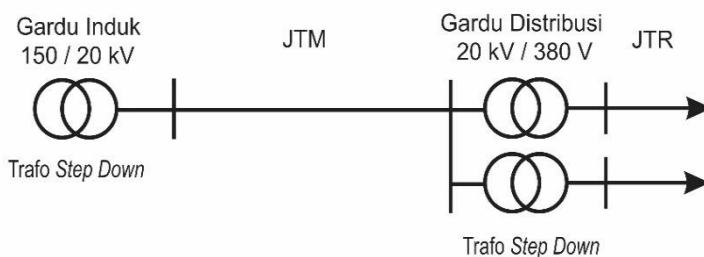
Oleh sebab itu Krisna Sadewa dan Iqbal Asad Kutsuro dalam mengerjakan tugas akhir ini membutuhkan teori penunjang diantaranya yakni :

2.2 Jaringan Distribusi Tegangan Menengah [1]

Bagian dari sistem tenaga listrik yang paling dekat dengan pelanggan adalah sistem distribusi. Sistem distribusi merupakan sistem yang paling banyak mengalami gangguan, sehingga masalah utama dalam operasi sistem distribusi pada Gambar 2.1.

Dilihat dari tegangannya sistem distribusi pada saat ini dapat dibedakan dalam 2 macam yakni

- Distribusi Primer, sering disebut sistem Jaringan Tegangan Menengah (JTM) dengan tegangan operasi nominal 20 kV/ 11,6 kV
- Distribusi Sekunder, sering disebut sistem Jaringan Tegangan Rendah (JTR) dengan tegangan operasi nominal 380 / 220 Volt



Gambar 2.1 Sistematika Jaringan Distribusi Tegangan Menengah

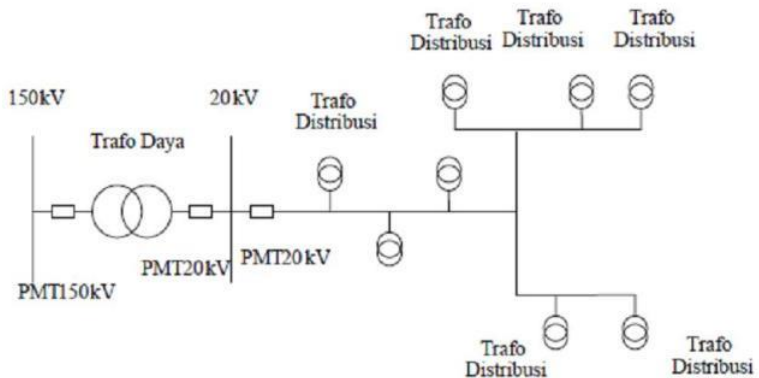
Jaringan distribusi terdiri dari distribusi primer dan sekunder. Jaringan distribusi primer menggunakan tegangan menengah dengan besar 20 kV dan jaringan distribusi sekunder menggunakan tegangan dengan besar 380 V.

2.2.1 Jenis Jaringan Distribusi Primer [1]

Terdapat 2 jenis jaringan distribusi primer yakni, sistem radial dan sistem loop.

1. Sistem Jaringan *Radial*

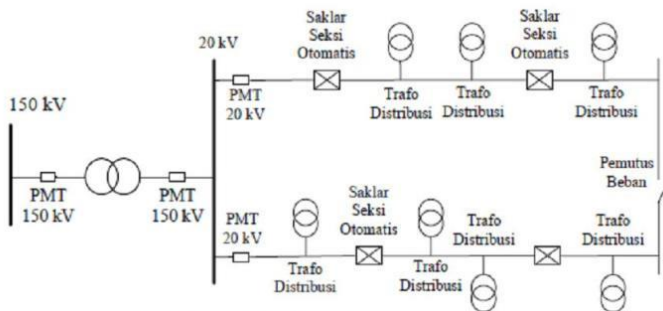
Sistem jaringan *radial* merupakan sistem paling sederhana karena terdiri dari penyulang (*feeder*) yang menuju langsung pada beban pada Gambar 2.2. *Feeder* dapat ditambahkan dengan trafo dan membentuk saluran samping atau lateral. Hubungan antara *feeder* dan saluran samping dibatas oleh sekering (*fuse*). Hal ini bertujuan bilamana terjadi gangguan dan sekering tidak bekerja, maka proteksi pada gardu induk akan bekerja, dan seluruh *feeder* akan kehilangan energi.



Gambar 2.2 Sistem Jaringan *Radial*

2. Sistem Jaringan *Loop*

Pada jaringan tegangan menengah struktur lingkaran (*Loop*). Gambar 2.3 adalah jaringan yang dimulai dari suatu titik pada rel daya yang berkeliling di daerah beban kemudian kembali ke titik rel daya semula. Pola ini ditandai pula dengan adanya dua sumber pengisian yakni sumber utama dan sebuah sumber cadangan. Jika salah satu sumber pengisian (saluran utama) mengalami gangguan, akan dapat digantikan oleh sumber pengisian yang lain (saluran cadangan). Jaringan PMT dengan pola ini biasa dipakai pada sistem distribusi yang melayani beban dengan kebutuhan kontinuitas pelayanan yang baik (lebih baik dari pola *radial*)



Gambar 2.3 Sistem Jaringan *Loop*

2.3 Gangguan Hubung Singkat [2]

Dalam operasi sistem tenaga listrik sering terjadi gangguan – gangguan yang dapat mengakibatkan terganggunya penyaluran tenaga listrik kekonsumen. Gangguan adalah penghalang dari suatu sistem yang sedang beroperasi atau suatu keadaan dari sistem penyaluran tenaga listrik yang menyimpang dari kondisi normal. Sesuai dengan SPLN No 2:1978 Pasal 9 bahwa dalam system distribusi listrik tiga fasa tiga kawat menggunakan pentanahan dengan tahanan sebagai suatu sistem distribusi yang berlaku untuk wilayah kerja PLN di seluruh Indonesia. Dalam Pasal 10 dijelaskan bahwa untuk sistem Jawa Timur yang menggunakan pentanahan dengan tahanan tingi maka nilai tahanan yang digunakan adalah 500 *Ohm* dengan arus gangguan hubung singkat satu fasa ke tanah maksimal 25 A jika menggunakan saluran udara. Pada sistem ini pula arus hubung singkat 2 fasa biasanya lebih kecil dari pada arus hubung singkat 3 fasa. Sedang arus gangguan 1 fasa ketanah hampir selalu lebih kecil daripada arus hubung singkat 3 fasa karena:

1. Umumnya impedansi urutan nolnya lebih besar dari pada impedansi urutan positif/ negatif.
2. Gangguan tanah melalui tahanan gangguan
3. Untuk pentanahan yang menggunakan tahanan, tahanan netralnya akan membatasi arus gangguan 1 fasa ketanah.

Kondisi sebaliknya terjadi (gangguan 1 fasa ke tanah lebih besar dari pada arus hubung singkat 3 fasa) apabila lokasi gangguan berada di pusat pembangkit atau dekat pusat pembangkit pada sistem dengan pentanahan langsung (*solid grounded*).

2.3.1 Penyebab Gangguan Hubung Singkat [2]

Gangguan biasanya diakibatkan oleh kegagalan isolasi di antara penghantar fasa atau antara penghantar fasa dengan tanah. Secara nyata kegagalan isolasi dapat menghasilkan efek pada sistem yakni menghasilkan arus yang cukup besar. Ada beberapa faktor yang menyebabkan terjadinya gangguan, antara lain sebagai berikut :

a. Faktor Manusia

Faktor ini terutama menyangkut kesalahan atau kelalaian dalam memberikan perlakuan pada sistem. Misalnya salah menyambungkan rangkaian, salah dalam mengkalibrasi suatu piranti pengaman, dan sebagainya.

b. Faktor Internal

Faktor ini menyangkut gangguan – gangguan yang berasal dari sistem itu sendiri. Misalnya usia pakai, keausan, dan sebagainya. Hal ini bisa mengurangi sensitivitas relai pengaman, juga mengurangi daya isolasi peralatan listrik lainnya.

c. Faktor Eksternal

Faktor ini meliputi gangguan – gangguan yang berasal dari lingkungan di sekitar sistem misalnya cuaca, gempa bumi, banjir, dan sambaran petir. Disamping itu ada kemungkinan gangguan dari binatang.

Gangguan-gangguan tersebut menyebabkan terjadinya:

1. Interupsi kontinuitas pelayanan daya kepada para konsumen apabila gangguan itu sampai menyebabkan terputusnya suatu rangkaian (*circuit*) atau menyebabkan keluarnya satu unit pembangkit.
2. Penurunan tegangan yang cukup besar menyebabkan rendahnya kualitas tenaga listrik dan merintangi kerja normal pada peralatan konsumen.

2.3.1 Jenis Gangguan Hubung Singkat [3]

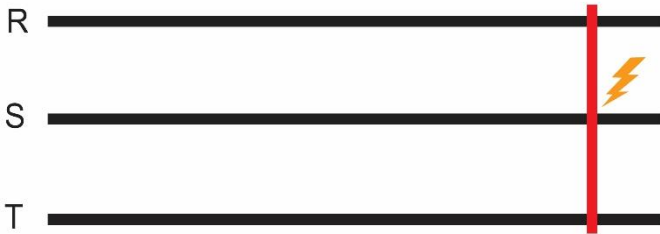
Gangguan pada sistem distribusi merupakan kejadian yang menyebabkan bekerjanya *relay* dan menjatuhkan pemutus tenaga yang melalui diluar kehendak operator, sehingga menyebabkan putusnya aliran daya yang melalui pemutus tersebut. berdasarkan sifatnya gangguan dibagi menjadi dua yakni gangguan temporer dan gangguan permanen.

Gangguan bersifat temporer ditandai dengan normalnya kerja PMT setelah dimasukan kembali dengan penyebab gangguan dapat diatasi

sendirinya, sedangkan yang bersifat permanen ditandai dengan tripnya *breaker*, dan penyebab gangguan harus mendapatkan pemeliharaan dari operator atau petugas lapangan sebelum akhirnya kembali seperti semula.

Beberapa jenis gangguan hubung singkat adalah:

1) Gangguan Hubung Singkat Tiga Fasa



Gambar 2.4 Gangguan Hubung Singkat Tiga Fasa

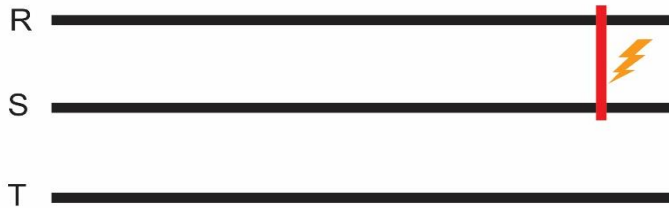
2) Gangguan Hubung Singkat Tiga Fasa ke Tanah



Gambar 2.5 Gangguan Hubung Singkat Tiga Fasa ke Tanah

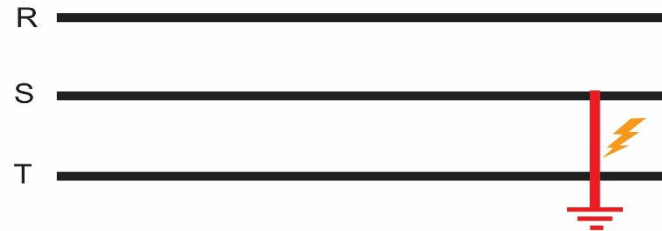
Jenis gangguan hubung singkat mempengaruhi besarnya arus hubung singkat yang terjadi, arus hubung singkat yang terjadi akibat tiga fasa pada Gambar 2.4 dan Gambar 2.5 mempunyai nilai arus hubung singkat yang lebih besar daripada gangguan yang lain. Gangguan hubung singkat tiga fasa ini biasanya diakibatkan oleh robohnya pohon ke jaringan yang menyebabkan ketiga fasa tersambung menjadi satu. Selain akibat yang dihasilkan lebih besar perbaikan yang dilakukan saat gangguan juga memerlukan perbaikan yang lebih lama dari gangguan yang lain.

3) Gangguan Hubung Singkat Dua Fasa



Gambar 2.6 Gangguan Hubung Singkat Dua Fasa

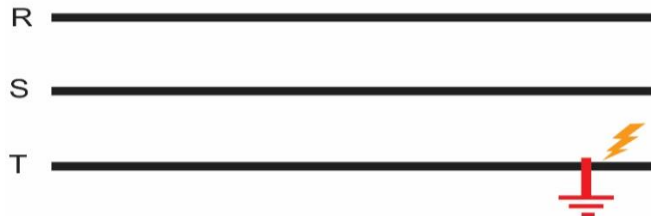
4) Gangguan Hubung Singkat Dua Fasa ke Tanah



Gambar 2.7 Gangguan Hubung Singkat Dua Fasa ke Tanah

Untuk jenis gangguan hubung singkat dua fasa pada Gambar 2.6 dan Gambar 2.7 biasanya terjadi akibat ranting pohon atau bisa juga dikarenakan adanya hewan yang merambat dari satu fasa ke fasa yang lainnya sehingga menyebabkan kedua fasa tersambung. Untuk penanganannya sendiri memerlukan waktu yang tidak begitu lama karena kerusakan yang terjadi tidak begitu parah dan hanya merupakan kerusakan sementara.

5) Gangguan Satu Fasa ke Tanah



Gambar 2.8 Gangguan Satu Fasa ke Tanah

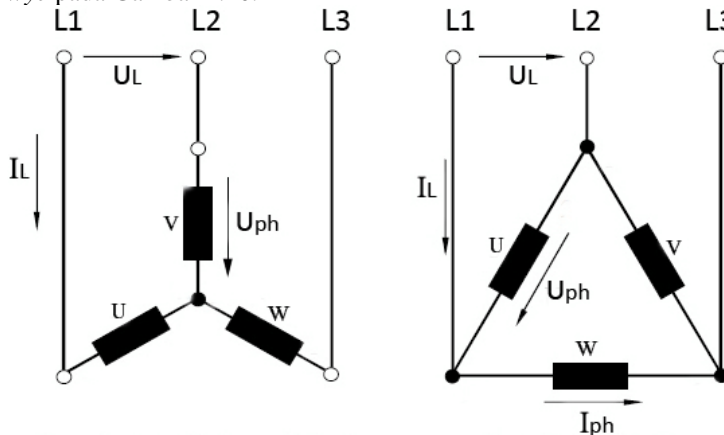
Untuk gangguan hubung singkat satu fasa ke tanah pada Gambar 2.8 mempunyai karakteristik yang hampir sama seperti gangguan hubung singkat dua fasa. Gangguan satu fasa ini merupakan jenis gangguan yang sering terjadi lebih banyak daripada gangguan yang lainnya.

2.4 *Transformator Step Down* [4]

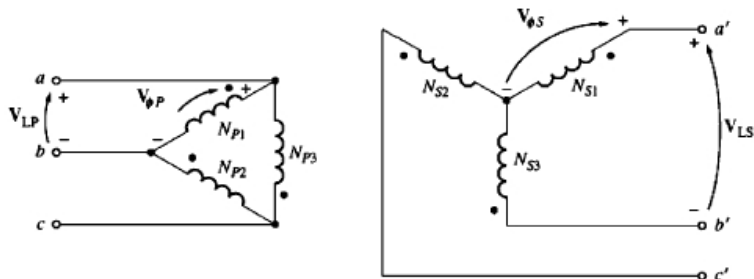
Trafo tiga fasa terdiri dari 3 *transformer*, baik disusun secara terpisah atau tergabung dalam satu inti. Sisi primer dan sekunder dari setiap trafo 3 fasa dapat dikoneksikan baik secara *wye* ataupun *delta*. hal ini menyebabkan ada 4 konfigurasi yang mungkin dilakukan yakni :

1. *Wye-wye*
2. *Wye-delta*
3. *Delta-wye*
4. *Delta-delta*

Dari beberapa konfigurasi yang sering digunakan dalam kehidupan sehari-hari yakni konfigurasi *wye-delta* dan konfigurasi *delta-wye*. Untuk konfigurasi *wye-delta* pada Gambar 2.9 biasanya sering dipakai pada konfigurasi saat *starting* motor industri, sedangkan untuk konfigurasi jaringan distribusi tegangan menengah adalah jaringan konfigurasi *delta-wye* pada Gambar 2.10.



Gambar 2.9 Konfigurasi *Wye-Delta*



Gambar 2.10 Konfigurasi *Delta-Wye*

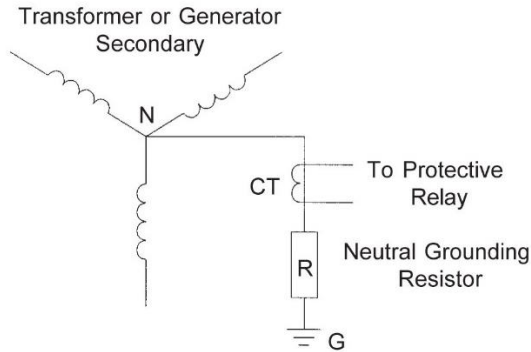
Dalam konfigurasi ini, tegangan pada sisi primer sama dengan tegangan per fasa $V_{L-L} = V_{L-N}$, sementara tegangan pada sisi sekunder adalah $V_{L-L} = \sqrt{3}V_{L-N}$, sehingga Persamaan rasio tegangan *line to line* untuk trafo dengan Persamaan 2.1 ini:

$$\frac{V_{LP}}{V_{LS}} = \frac{V_{\phi P}}{\sqrt{3}V_{\phi S}} \dots\dots\dots(2.1)$$

Sistem tiga fasa 3 kawat umumnya diterapkan pada jaringan distribusi primer yang menggunakan tegangan menengah. Masing – masing fase diberi suatu tanda yakni R atau U dengan warna merah, S atau V dengan warna kuning, dan T atau W dengan warna biru.

2.5 NGR (*Neutral Grounding Resistance*) [5]

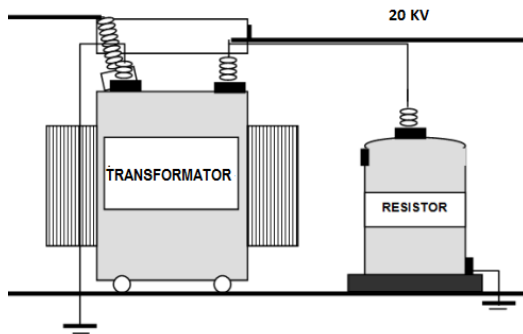
Neutral Grounding Resistance atau NGR adalah salah satu metode pentanahan Trafo Tenaga dengan menggunakan *resistor*. NGR adalah sebuah tahanan yang dipasang serial dengan netral sekunder pada *transformator* sebelum terhubung ke ground/tanah pada Gambar 2.11. Tujuan dipasangnya NGR adalah untuk mengontrol besarnya arus gangguan yang mengalir dari sisi netral ke tanah. Selain itu *Neutral Grounding Resistor* berfungsi sebagai pembatas arus dalam saluran netral trafo. Agar NGR dapat berfungsi sesuai desainnya perlu dipastikan bahwa nilai tahanan dari NGR tersebut sesuai dengan spesifikasinya dan tidak mengalami kerusakan. Hal ini terkait dengan pola pengamanan trafo pada sistem pentanahan saat terjadinya gangguan yang besar yang dapat membahayakan trafo.



Gambar 2.11 *Neutral Grounding Resistance*

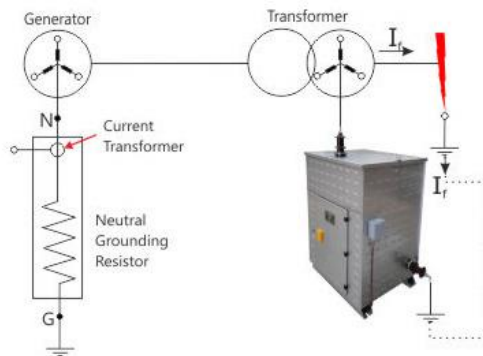
Ada dua jenis NGR yang biasa digunakan yakni,

1. NGR *Liquid* berarti *resistornya* menggunakan larutan air murni yang ditampung didalam bejana pada Gambar 2.12 dan ditambahkan garam (NaCl) untuk mendapatkan nilai resistansi yang diinginkan.



Gambar 2.12 *Neutral Grounding Liquid*

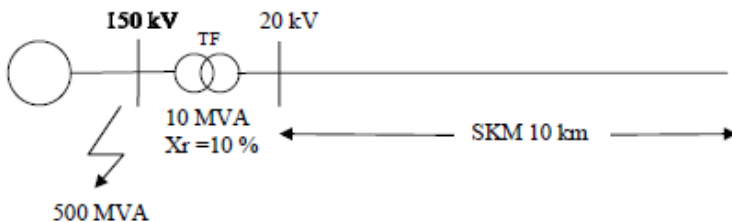
2. NGR *Solid* adalah NGR jenis padat terbuat dari *Stainless Steel*, *FeCrAl*, *Cast Iron*, *Copper Nickel* atau *Nichrome* pada Gambar 2.13 yang diatur sesuai nilai tahanannya.



Gambar 2.13 *Neutral Grounding Resistance Solid*

2.6 Perhitungan Arus Hubung Singkat [6]

Pada perhitungan arus hubung singkat ini kita akan mencari nilai arus ubung singkat dengan contoh perhitungan dalam kondisi real. Dengan dilakukannya perancangan arus hubung sngkat nantinya kita akan mengetahui berapa nilai arus hubung singkat yang akan terjadi pada modul gangguan yang akan kita buat. Contoh perhitungan akan diumpamakan pada suatu gardu induk (GI) yang terpasang satu trafo tenaga 150/20 kV dengan daya sebesar 10 MVA dengan Impedansi = 10%, netral trafo tenaga ini ditanahkan melalui tahanan 40 *Ohm*. *Short Circuit* level pada bus 150 kV pada Gambar 2.14.



Gambar 2.14 Skema Perancangan Arus Hubung Singkat

2.6.1 Menghitung Impedansi Sumber [6]

Dari Gambar 2.14 diketahui bahwa data hubung singkat di bus 150 kV gardu induk (GI) adalah sebesar 500 MVA dengan Persamaan 2.2.

$$X_s = \frac{KV^2}{MVA} \dots\dots\dots(2.2)$$

$$X_s = \frac{150^2}{500}$$

$$X_s = 45 \text{ Ohm}$$

Dari perhitungan di atas didapat nilai impedansi sumber di sisi 150 kV, karena arus gangguan hubung singkat yang akan dihitung adalah gangguan hubung singkat di sisi 20 kV, maka impedansi tersebut harus dikonversasikan dulu ke sisi 20 KV. Untuk mengkonversikan impedansi yang terletak di sisi 150 kV ke sisi 20 kV dapat dilakukan dengan Persamaan 2.3.

$$X_s \text{ 20 kV} = \frac{20^2}{150^2} \dots\dots\dots(2.3)$$

$$X_s = 0,8 \text{ Ohm}$$

2.6.2 Menghitung Reaktansi Trafo [6]

Reaktansi trafo tenaga sisi 10 MVA adalah sebesar 10%. Untuk mencari nilainya dalam *Ohm* dihitung dengan cara mencari nilai *Ohm* 100% pada trafo 10 MVA dengan Persamaan 2.4.

$$X_t = \frac{KV^2}{MVA} \dots\dots\dots(2.4)$$

$$X_t = \frac{20^2}{10}$$

$$X_t = 40 \text{ Ohm}$$

Dari hasil perhitungan pada Persamaan 2.4 maka dapat dicari reaktansi urutan positif dan negatif sebesar 10% dari total keseluruhan reaktansi trafo. Maka nilai reaktansi positif dan negatif trafo didapat dari Persamaan 2.5.

$$X_{positif} = X_{negatif} \dots\dots\dots(2.5)$$

$$X_t = 10\% \times 40$$

$$X_t = 4 \text{ Ohm}$$

2.6.3 Menghitung Impedansi Penyulang [6]

Selanjutnya kita akan menghitung besar impedansi penyulang. Besar nilai impedansi penyulang tergantung dari besarnya impedansi per km dari penyulang yang akan dihitung, dimana besar nilainya ditentukan dari konfigurasi tiang yang digunakan untuk jaringan SUTM.

Pada contoh perhitungan, akan dimisalkan dengan nilai $Z = (R + jX)$ Ohm/km senilai Persamaan 2.6 dan Persamaan 2.7.

$$Z_1 = Z_2 = (0,12 + j0,23)/\text{km} \dots \dots \dots (2.6)$$

$$Z_0 = (0,18 + j0,53)/\text{km} \dots \dots \dots (2.7)$$

Dengan demikian nilai impedansi penyulang untuk lokasi gangguan dalam perhitungan ini diperkirakan terjadi pada lokasi dengan jarak 25%, 50%, 75%, 100% dari panjang penyulang, sehingga dapat dihitung pada Tabel 2.1 dan Tabel 2.2.

Tabel 2.1 Perhitungan Impedansi Urutan Positif dan Negatif

Impedansi Urutan Positif dan Negatif	
% Panjang	Impedansi Penyulang (Z_1, Z_2)
25	$25\% \times 10 \times (0,12 + j0,23) = 0,3 + j0,575$
50	$50\% \times 10 \times (0,12 + j0,23) = 0,6 + j1,15$
75	$75\% \times 10 \times (0,12 + j0,23) = 0,9 + j1,725$
100	$100\% \times 10 \times (0,12 + j0,23) = 1,2 + j2,3$

Tabel 2.2 Perhitungan Impedansi Urutan Nol

Impedansi Urutan Positif dan Negatif	
% Panjang	Impedansi Penyulang (Z_1, Z_2)
25	$25\% \times 10 \times (0,18 + j0,53) = 0,45 + j1,325$
50	$50\% \times 10 \times (0,18 + j0,53) = 0,90 + j2,65$
75	$75\% \times 10 \times (0,18 + j0,53) = 1,35 + j3,975$
100	$100\% \times 10 \times (0,18 + j0,53) = 1,83 + j5,30$

2.6.4 Menghitung Impedansi Ekivalen Jaringan [6]

Perhitungan yang akan dilakukan adalah perhitungan besarnya nilai impedansi ekivalen dari nilai positif (Z_{1eq}), negatif (Z_{2eq}) dan nol (Z_{0eq}) dari titik gangguan sampai ke sumber.

Karena dari sumber ke titik gangguan impedansi yang terbentuk adalah seri, maka perhitungan Z_{1eq} dan Z_{2eq} dapat dilakukan dengan cara langsung menjumlahkan impedansi-impedansi tersebut, sedangkan untuk perhitungan Z_{0eq} dimulai dari titik gangguan sampai ke trafo tenaga yang netralnya ditanahkan.

Untuk perhitungan Z_{1eq} dan Z_{2eq} dilakukan dengan Persamaan 2.8.

$$\begin{aligned} Z_{1eq} &= Z_{2eq} = Z_{s1} + Z_{t1} + Z_1 \text{ penyulang} \dots \dots \dots (2.8) \\ Z_{1eq} &= Z_{2eq} = j0,8 + j4,0 + Z_1 \text{ penyulang} \\ Z_{1eq} &= Z_{2eq} = j4,8 + Z_1 \text{ penyulang} \end{aligned}$$

Karena lokasi gangguan diasumsikan terjadi pada 25%, 50%, 75%, dan 100% panjang Penyulang, maka Z_{1eq} dan Z_{2eq} yang didapat pada Tabel 2.3.

Tabel 2.3 Perhitungan Ekuivalen Positif dan Negatif

Impedansi Urutan Positif dan Negatif	
% Panjang	Impedansi Ekuivalen (Z_1, Z_2)
25	$j4,8 + (0,3 + j0,575) = 0,3 + j5,375$
50	$j4,8 + (0,6 + j1,150) = 0,6 + j5,950$
75	$j4,8 + (0,9 + j1,725) = 0,9 + j6,525$
100	$j4,8 + (1,2 + j2,300) = 1,2 + j7,100$

Untuk menghitung Impedansi Z_{0eq} , maka diumpamakan trafo tenaga yang terpasang mempunyai hubungan *wye-delta*, dimana mempunyai nilai sebesar 12 *Ohm*, dan nilai tahanan pentahanan 120 *Ohm*. Untuk perhitungan nilai Z_{0eq} dilakukan dengan Persamaan 2.9.

$$\begin{aligned} Z_{0eq} &= Z_{t0} + 3R_n + Z_0 \text{ penyulang} \dots \dots \dots (2.9) \\ Z_{0eq} &= j12 + 120 + Z_0 \text{ penyulang} \end{aligned}$$

Karena lokasi gangguan diasumsikan terjadi pada 25%, 50%, 75%, dan 100% panjang Penyulang, maka Z_{0eq} yang didapat pada Tabel 2.4.

Tabel 2.4 Perhitungan Ekvivalen Urutan Nol

Impedansi Urutan Positif dan Negatif	
% Panjang	Impedansi Ekvivalen (Z1,Z2)
25	$j12 + 120 + (0,45 + j1,325) = 120,45 + j13,325$
50	$j12 + 120 + (0,90 + j2,650) = 120,90 + j14,650$
75	$j12 + 120 + (1,35 + j3,975) = 121,35 + j15,975$
100	$j12 + 120 + (1,80 + j5,300) = 121,80 + j17,300$

2.6.5 Menghitung Arus Gangguan Antar Fasa [6]

Persamaan dasar yang digunakan pada perhitungan arus gangguan hubung singkat 2 fasa ini adalah Persamaan 2.10.

$$I = \frac{V_{ph-ph}}{Z} \dots\dots\dots (2.10)$$

di mana :

- I = Arus gangguan hubung singkat
- V_{ph-ph} = Tegangan fasa-fasa sistem 20 kV
- Z = Impedansi

Sehingga dari Persamaan 2.10 dapat diketahui bahwa arus hubung singkat dapat dihitung dengan Persamaan 2.11.

$$I = \frac{V_{ph-ph}}{2 \times Z1_{eq}} \dots\dots\dots (2.11)$$

Gangguan hubung singkat 2 fasa juga dihitung untuk lokasi gangguan yang diasumsikan terjadi pada 25%, 50%, 75%, dan 100% dari panjang penyulang. Dalam hal ini dianggap nilai $Z1_{eq} = Z2_{eq}$, sehingga Persamaan arus gangguan hubung singkat 2 Fasa dapat disederhanakan menjadi Persamaan 2.11.

Perhitungan arus hubung singkat dapat dilihat pada Tabel 2.5.

Tabel 2.5 Perhitungan Arus Hubung Singkat 2 Fasa

No	Panjang (%)	Perhitungan	Arus Gangguan
1	25	$\frac{20.000}{2 \times (0,3 + j5,375)}$	1857,6 A
2	50	$\frac{20.000}{2 \times (0,6 + j5,950)}$	1672,2 A
3	75	$\frac{20.000}{2 \times (0,9 + j6,525)}$	1518,2 A
4	100	$\frac{20.000}{2 \times (1,2 + j7,100)}$	1388,8 A

Dari hasil perhitungan arus hubung singkat 2 fasa pada Tabel 2.5 dapat diambil kesimpulan bahwa besar arus hubung singkat 2 fasa semakin jauh dari sumber atau dari GI maka akan semakin kecil arus gangguan hubung singkatnya. Hal ini dikarenakan faktor besar resistansi dan impedansi jaringan sangat mempengaruhi arus hubung singkat yang terjadi. Semakin panjang jarak gangguan maka nilai impedansi akan semakin bertambah juga sehingga semakin besar nilai impedansi maka akan memperkecil nilai arus hubung singkat yang terjadi. Sehingga bila nilai arus hubung singkat yang terjadi semakin kecil maka akan lebih sulit untuk dideteksi letak gangguan hubung singkat yang terjadi.

2.6.6 Menghitung Arus Gangguan Satu Fasa ke Tanah [6]

Persamaan dasar yang digunakan pada perhitungan arus gangguan hubung singkat fasa ke tanah ini adalah Persamaan 2.12.

$$I = \frac{V_{ph-n}}{Z} \dots\dots\dots (2.12)$$

di mana :

- I = Arus gangguan hubung singkat
- V_{ph-n} = Tegangan fasa-netral sistem 20 kV
- Z = Impedansi

Sehingga dari Persamaan 2.12 dapat diketahui bahwa arus hubung singkat dapat dihitung dengan Persamaan 2.13.

$$I = \frac{V_{ph-n}}{2 \times Z1_{eq} + Z0_{eq}} \dots\dots\dots (2.13)$$

Seperti gangguan hubung singkat 2 fasa, Gangguan hubung singkat fasa ke tanah juga dihitung untuk lokasi gangguan yang diasumsikan terjadi pada 25%, 50%, 75%, dan 100% dari panjang penyulang. Dalam hal ini dianggap nilai $Z_{1eq} = Z_{2eq}$ kemudian ditambahkan dengan Z_0 eq, sehingga Persamaan arus gangguan hubung singkat fasa ke tanah di atas dapat disederhanakan menjadi Persamaan 2.13.

Perhitungan arus hubung singkat adalah sebagai berikut:

Tabel 2.6 Perhitungan Arus Hubung Singkat Fasa ke Tanah

No	Panjang (%)	Perhitungan	Arus Gangguan
1	25	$\frac{34641,016}{2 \times (0,3 + j5,375) + (120,45 + j13,325)}$	280,74 A
2	50	$\frac{34641,016}{2 \times (0,6 + j5,950) + (120,9 + j14,65)}$	277,23 A
3	75	$\frac{34641,016}{2 \times (0,9 + j6,525) + (120,45 + j15,975)}$	273,8 A
4	100	$\frac{34641,016}{2 \times (1,2 + j7,10) + (120,45 + j17,30)}$	270,4 A

Dari hasil perhitungan arus hubung singkat fasa ke tanah pada Tabel 2.6 dapat diambil kesimpulan bahwa besar arus hubung singkat semakin jauh dari sumber atau dari GI maka akan semakin kecil arus gangguan hubung singkatnya. Hal ini dikarenakan faktor besar resistansi dan impedansi jaringan sangat mempengaruhi arus hubung singkat yang terjadi. Semakin panjang jarak gangguan maka nilai impedansi akan semakin bertambah juga sehingga semakin besar nilai impedansi maka akan memperkecil nilai arus hubung singkat yang terjadi. Sehingga bila nilai arus hubung singkat yang terjadi semakin kecil maka akan lebih sulit untuk dideteksi letak gangguan hubung singkat yang terjadi.

2.7 Sensor Arus YHDC ST-013 [7]

Sensor arus merupakan sebuah alat yang digunakan untuk mendeteksi besar arus pada sebuah rangkaian listrik. Sensor tegangan yang digunakan adalah sensor tegangan YHDC- ST013 pada Gambar 2.15. Sensor arus ini dapat mendeteksi arus AC atau DC hingga maksimal 30 A dan keluarannya adalah tegangan AC 1 V yang memudahkan penggunaannya sehingga bisa langsung digunakan ke perangkat pendeteksi

atau monitoring arus. Sensor arus YHDC ST-013 ini mempunyai spesifikasi pada Tabel 2.7.

Tabel 2.7 Spesifikasi Sensor Arus YHDC-013

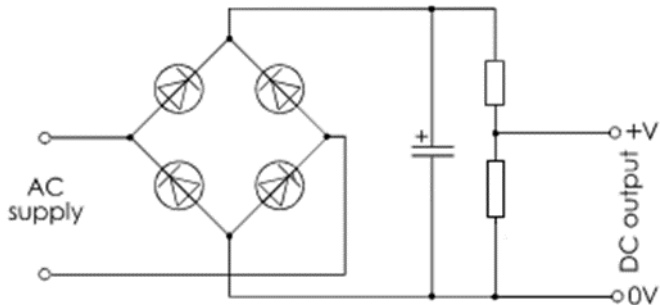
Arus input	Tegangan output
Max 30 A	Max 1 V



Gambar 2.15 Sensor Arus YHDC ST-013

2.8 Sensor Tegangan [7]

Sensor tegangan merupakan alat yang digunakan untuk mendeteksi besar tegangan yang melalui suatu peralatan listrik. Sensor tegangan menggunakan rangkaian pembagi tegangan yang telah diberi penyearah terlebih dahulu. Penggunaan rangkaian pembagi tegangan dinilai lebih efisien dan juga lebih tepat karena digunakan untuk mengukur tegangan yang bernilai sangat kecil. Tegangan AC yang akan diukur harus disearahkan terlebih dahulu menggunakan *rectifier* pada Gambar 2.16 agar menjadi sinyal DC. Rangkaian ini menggunakan kapasitor $10\mu\text{F}$ 50 Volt dengan *resistor* 10K $\frac{1}{4}$ Watt, dan juga *multiturn* 100 K sebagai komponen pengatur nilai resistansi. Fungsi *multiturn* 10K ini sebagai alat pengatur tegangan yang telah disearahkan oleh *rectifier* sisir 5 A yang terletak sebelum rangkaian pembagi tegangan, pengaturan tegangan ini bertujuan agar bisa langsung diolah oleh mikrokontroler.



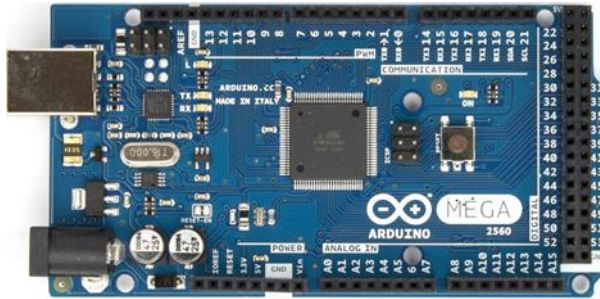
Gambar 2.16 Rangkaian Penyearah Gelombang Penuh

2.9 Board Arduino Mega [8]

Board Arduino Mega adalah sebuah sistem minimum berbasis mikrokontroler ATmega2560. Spesifikasi *board* Arduino berdasarkan *datasheet* adalah sebagai berikut :

1. Tegangan aktif 5V
2. Tegangan masukan 7-12V
3. Tegangan masukan (batas) 6-20V
4. Pin *Input/Output Digital* 54 (6 dapat digunakan sebagai PWM)
5. Pin *Input Analog* 16
6. Arus DC pin I/O 40 mA
7. Arus DC pada 3,3V 50 mA
8. *Flash Memory* 128 KB (ATmega328), 0,5 KB digunakan *bootloader*
9. SRAM 8 KB (ATmega328)
10. EEPROM 4 KB (ATmega328)
11. *Clock Speed* 16 MHz

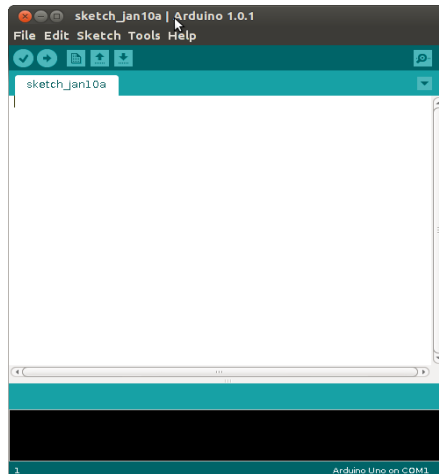
Board ini dilengkapi *jack* DC dengan ukuran tegangan antara 6-20 Volt. Selain itu, *board* ini juga dilengkapi USB untuk komunikasi serial sehingga dapat dengan mudah dikoneksikan dengan PC. Bentuk fisik *board* Arduino pada Gambar 2.17.



Gambar 2.17 Board Arduino Mega

2.10 Arduino IDE [8]

Board Arduino dapat di program menggunakan *software open source* bawaan Arduino IDE. Arduino IDE adalah sebuah aplikasi *crossplatform* yang berbasis Bahasa pemrograman *Processing* dan *Wiring*. Arduino IDE di desain untuk mempermudah pemrograman dengan adanya kode editor yang dilengkapi dengan *syntax highlighting*, *brace matching*, dan indentasi otomatis untuk kemudahan pembacaan program, serta dapat melakukan proses *compile* dan *upload* program ke *board* dalam satu klik. Jendela Arduino IDE dapat dilihat pada Gambar 2.18.



Gambar 2.18 Jendela Arduino IDE

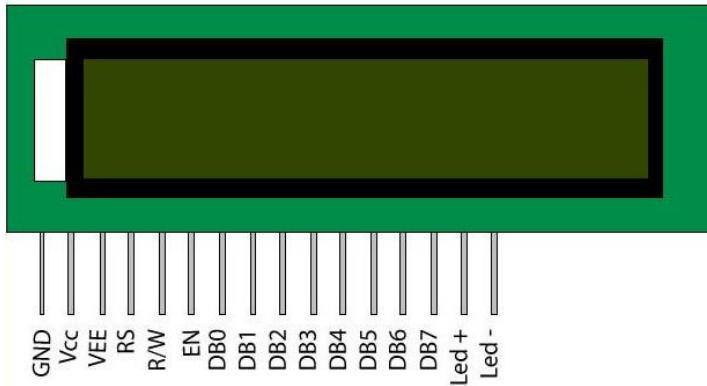
IDE Arduino adalah *software* yang sangat canggih ditulis dengan menggunakan Java. IDE Arduino terdiri dari:

1. Editor program, sebuah *window* yang memungkinkan pengguna menulis dan mengedit program dalam bahasa *Processing*.
2. *Compiler*, sebuah modul yang mengubah kode program (bahasa *Processing*) menjadi kode biner. Bagaimanapun sebuah mikrokontroler tidak akan bisa memahami bahasa *Processing*. Yang bisa dipahami oleh mikrokontroler adalah kode biner. Itulah sebabnya *compiler* diperlukan dalam hal ini.
3. *Uploader*, sebuah modul yang memuat kode biner dari komputer ke dalam *memory* di dalam papan Arduino.

2.11 LCD (*Liquid Crystal Display*) [9]

LCD (*Liquid Crystal Display*) bisa memunculkan gambar atau tulisan dikarenakan terdapat banyak sekali titik cahaya yang terdiri dari satu buah kristal cair sebagai sebuah titik cahaya. Fungsi pin yang terdapat pada LCD terlihat pada Gambar 2.19. Modul LCD memiliki karakteristik:

1. Terdapat 16 x 2 karakter huruf yang bisa ditampilkan
2. Terdapat 192 macam karakter
3. Terdapat 80 x 8 *bit display* RAM (maksimal 80 karakter)
4. Memiliki kemampuan penulisan dengan 8 *bit* maupun dengan 4 *bit*
5. Dibangun dengan *osilator local*
6. Satu sumber tegangan 5 *Volt*
7. Otomatis *reset* saat tegangan dihidupkan
8. Bekerja pada suhu 0°C sampai 55°C



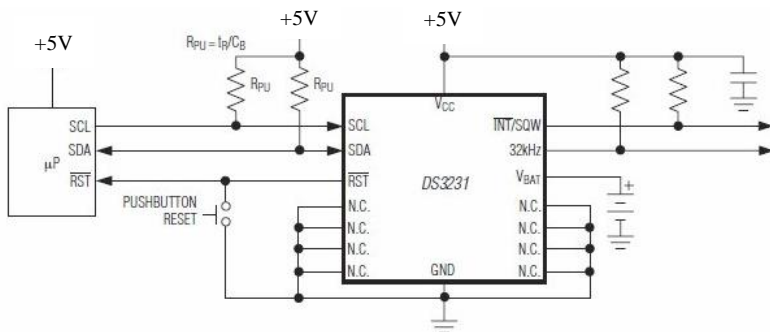
Gambar 2.19 LCD (*Liquid Crystal Display*)

2.12 Real Time Clock (RTC) [9]

Real Time Clock (RTC) merupakan suatu *chip IC* yang memiliki fungsi untuk mempertahankan detik, menit, jam, hari, tanggal, bulan, dan informasi tahun. Tanggal pada akhir bulan secara otomatis disesuaikan dengan bulan dengan kurang dari 31 hari, termasuk koreksi untuk tahun. Jam beroperasi baik dalam 24 jam atau format 12 jam dengan indikator AM/PM aktif ketika kondisi *low*. Sebuah temperature yang presisi dikompensasi sebagai tegangan referensi. Rangkaian komparator memonitor status VCC untuk mendeteksi gangguan listrik, untuk memberikan *output* ulang dan secara otomatis terhubung ke sumber cadangan jika diperlukan. Selain itu, pin RST akan aktif ketika tegangan 0 Volt yang dideteksi sebagai tombol tekan untuk menghasilkan perintah *reset*. Data-data yang tersimpan pada IC DS3231 disimpan pada *register* 00H untuk detik, 01H untuk menit, 02H untuk jam, 03H untuk hari, 04H untuk tanggal, 05H untuk bulan, 06H untuk tahun, 07H untuk control dan RAM 56x8 pada *register* 08H-3FH. *Register* tersebut bisa diakses oleh mikrokontroler melalui *bus* I2C.M. Modul RTC dapat dilihat pada Gambar 2.20 dan Skematik dari RTC dapat dilihat pada Gambar 2.21.



Gambar 2.20 Modul RTC



Gambar 2.21 Skematik RTC

2.13 SIM 900 atau GPRS dan GSM Shield [10]

Modul SIM900 atau GSM dan GPRS *Shield* adalah komponen yang sangat diperlukan untuk komunikasi antara sistem dengan *web server*, ini dikarenakan fungsi modul SIM900 sebagai pengirim data dari sistem ke *web server*. Spesifikasi dari SIM900 terdapat pada Tabel 2.8, dan *board* SIM900 terdapat pada Gambar 2.22.

AT Command adalah perintah yang dapat diberikan modem GSM/CDMA seperti untuk mengirim dan menerima data berbasis GSM/GPRS, atau mengirim dan menerima SMS. Modul SIM900 atau GSM dan GPRS *Shield* dikendalikan melalui perintah AT (GSM 07.07, 07.05, dan SIMCOM).

Tabel 2.8 Spesifikasi SIM 900

Spesifikasi	
Ukuran	77,2mm x 66,0mm x 1,6mm
Indikator	PWR, status LED, net status LED
Power Supply	9~20 Volt, compatible with Arduino
Protokol Komunikasi	UART
RoHS	Yes



Gambar 2.22 Modul SIM 900

2.14 Modem *Maestro 100* GSM/GPRS Serial [10]

Modem *Maestro 100* GSM/GPRS Serial adalah komponen yang digunakan untuk komunikasi antara sistem dengan komputer *server*, fungsi dari Modem *Maestro 100* ini adalah sebagai penerima data dari sistem ke komputer *server*.

Maestro 100 merupakan modem GSM siap pakai untuk layanan *voice*, data, faks dan SMS. Ini juga mendukung GPRS Class 10 untuk transfer data berkecepatan tinggi. *Maestro 100* dapat dengan mudah dikontrol dengan menggunakan *AT Command* untuk semua jenis operasi. Dengan port RS232 9 pin standar dan konektor audio seperti telepon (via

kabel opsional). Spesifikasi dari Modem *Maestro* 100 terdapat pada Tabel 2.9, Modem *Maestro* 100 terdapat pada Gambar 2.23.

Tabel 2.9 Spesifikasi Modem *Maestro* 100

Spesifikasi	
Ukuran/Berat	62.8mm x 60mm x 21.8mm/90gram
Indikator	<i>Operating Status LED</i>
Input Voltage	6~32 Volt DC
Interface	RS-232, SIM, Micro-FIT™ 4-pin connector, SMA antenna connector
Range Suhu Kerja (°C)	-30 ~ +70

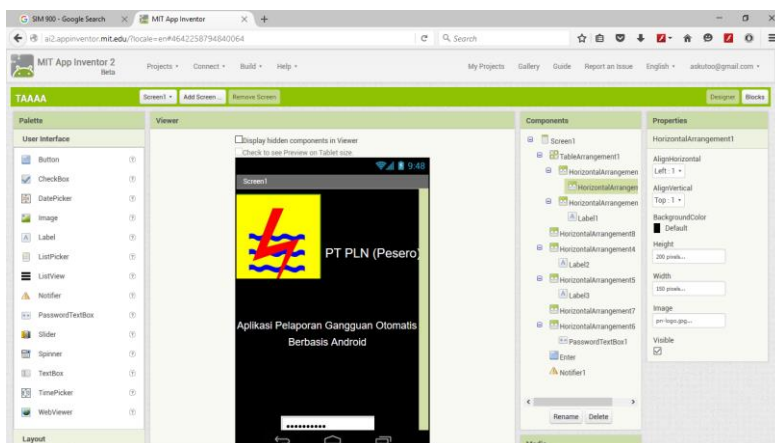


Gambar 2.23 Modem *Maestro* 100

2.15 MIT App Inventor [11]

MIT App Inventor adalah sebuah *tool* yang digunakan untuk membuat aplikasi android yang dikembangkan *Massachusetts Institute of Technology* (MIT). *Tool* ini berbasis *visual block programming* yang dalam penggunaannya hanya dengan menyusun “blok” yang merupakan bagian dari simbol-simbol perintah. *Framework visual programming* ini

terkait dengan Bahasa pemrograman *Scratch* dari MIT, yang secara spesifik merupakan implementasi dari *Open Block* yang didistribusikan oleh MIT *Scheller Teacher Education Program* yg diambil dari riset yang dilakukan oleh Ricarose Roque. *App Inventor* memungkinkan pengguna baru untuk memprogram komputer untuk menciptakan aplikasi perangkat lunak bagi sistem operasi Android. *App Inventor* menggunakan antarmuka grafis Gambar 2.24, serupa dengan antarmuka pengguna pada *Scratch* dan StarLogo TNG, yang memungkinkan pengguna untuk melakukan proses *drag* dan *drop* objek *visual* untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. Dalam menciptakan *App Inventor*, Google telah melakukan riset yang berhubungan dengan komputasi edukasional dan menyelesaikan lingkungan pengembangan *online* Google.

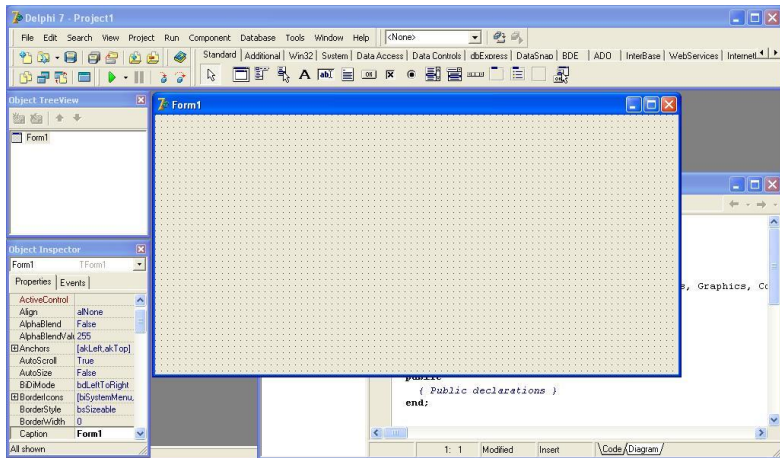


Gambar 2.24 Antarmuka *MIT App Inventor*

2.16 Delphi [11]

Delphi adalah sebuah *software* atau perangkat lunak yang digunakan untuk mengembangkan aplikasi konsol, desktop, web, ataupun perangkat mobile. Produk ini pada awalnya dikembangkan oleh CodeGear sebagai divisi pengembangan perangkat lunak milik Embarcadero, divisi tersebut sebelumnya adalah milik Borland. Bahasa *Delphi*, atau dikenal pula sebagai *object pascal* (pascal dengan ekstensi pemrograman berorientasi objek (PBO/OOP)) pada mulanya ditujukan hanya untuk Microsoft Windows, namun saat ini telah mampu digunakan

untuk mengembangkan aplikasi untuk Mac OS X, iOS, Microsoft .NET framework. Bahasa pemrograman yang digunakan oleh *Delphi* umumnya adalah bahasa pemrograman pascal. Berikut ini adalah tampilan awal *Delphi* pada Gambar 2.25.



Gambar 2.25 Jendela *Design Delphi*

-----*Halaman ini sengaja dikosongkan*-----

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Pada bab ini akan dibahas mengenai perancangan dan pembuatan alat “Desain dan Prototipe Sistem Deteksi Gangguan Impedansi Tinggi pada SUTM 20 kV dengan Sistem Pelaporan Gangguan Menggunakan Komunikasi Data Modul GSM” yang dilakukan oleh Krisna Sadewa dan Iqbal Asad Kutsuro.

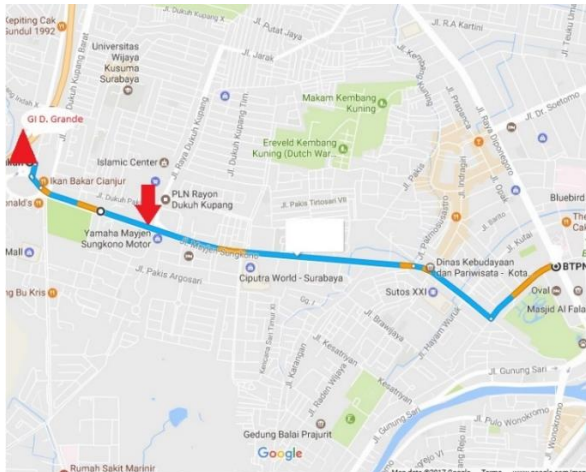
Perancangan dan pembuatan alat tugas akhir ini dibagi menjadi tiga tahap yang dikerjakan oleh Iqbal Asad Kutsuro pada Sub Bab 3.1 Perancangan Sistem Keseluruhan, Sub Bab 3.3 Perancangan Perangkat Keras (*hardware*) meliputi Sub Bab 3.2.2 Panel Deteksi Gangguan, Sub Bab 3.3.5 Sensor Arus dan Sub Bab 3.3.6 Sensor Tegangan.

Krisna Sadewa pada Sub Bab 3.4 Perancangan Perangkat Lunak (*Software*) yang meliputi Sub Bab 3.4.1 Pemrograman Sistem Arduino IDE, Sub Bab 3.4.2 Pemrograman pada *MIT App Inventor* dan Sub Bab 3.4.3 Pemrograman *Software* Menggunakan *Delphi*.

3.1 Perancangan Sistem Keseluruhan

Perancangan alat tugas akhir Krisna Sadewa dan Iqbal Asad Kutsuro mengambil contoh kondisi sebuah penyulang yang ada di Surabaya pada Gambar 3.1 yang merupakan gambar Penyulang Mayjend Sungkono yang menjadi dasar dari pembuatan tugas akhir ini, berdasarkan Data Induk Jaringan (DIJ) dari UP PLN Rayon Dukuh Kupang didapat data sebagai berikut :

Nama Penyulang	: Penyulang Mayjend Sungkono
Panjang Penyulang	: 4,1 km
Jenis Konduktor	: AAAC (150 mm ²)
Keterangan lain	: Sumber dari GI Darmo Grande



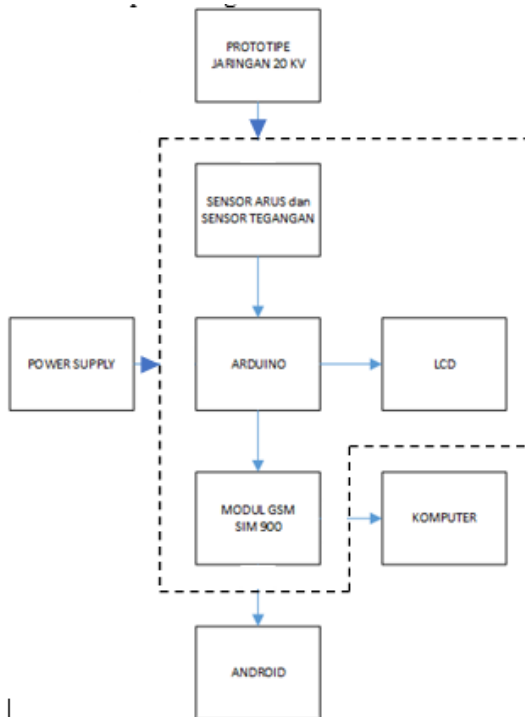
Gambar 3.1 Penyulang Mayjend Sungkono

Dengan adanya perancangan sistem keseluruhan dari penyulang Mayjend Sungkono, nantiya akan dibuat sebagai dasar dari perancangan dan penggambaran lokasi terjadinya gangguan pada keadaan yang sebenarnya dengan spesifikasi panjang penyulang 4 KM yang menggunakan kabel tanpa isolasi (AAAC) dengan diameter 150 mm² yang di *supply* dari GI Darmo Grande.

Pada tugas akhir ini gangguan pada sistem adalah gangguan satu fasa ke tanah dan gangguan antar fasa yang akan menyebabkan perubahan arus dan tegangan yang nantinya akan dideteksi oleh sensor arus dan sensor tegangan yang nantinya akan diolah oleh mikrokontroler dan ditampilkan pada LCD panel. Selain itu juga akan dikirimkan informasi berupa SMS ke *smartphone* dan juga ke komputer *server*.

Perancangan sistem secara keseluruhan dalam pembuatan alat yang digunakan oleh Krisna Sadewa dan Iqbal Asad Kutsuro diperlihatkan dengan diagram fungsional sistem pada Gambar 3.2.

Perancangan sistem secara keseluruhan dalam pembuatan alat yang digunakan oleh Krisna Sadewa dan Iqbal Asad Kutsuro diperlihatkan dengan pembagian modul simulasi gangguan untuk Iqbal Asad Kutsuro pada Gambar 3.3 dan Gambar 3.4 dan panel deteksi gangguan untuk Krisna Sadewa pada Gambar 3.6.



Gambar 3.2 Diagram Fungsional Sistem

3.2 Perancangan Mekanik

Pada perancangan mekanik tugas akhir ini terbagi menjadi tiga yakni perancangan Modul Gangguan, Panel *Box* (*Hardware*) dan perancangan *Software*.

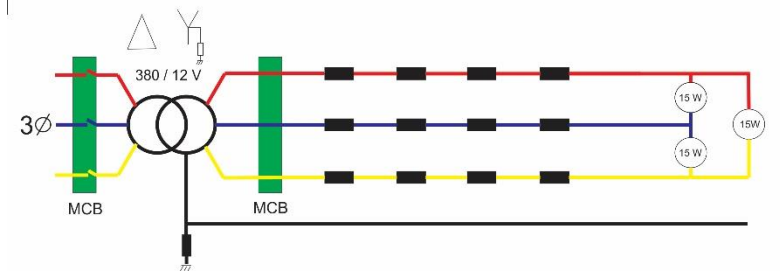
Perancangan Modul Gangguan dalam pembuatan alat yang dikerjakan oleh Iqbal Asad Kutsuro diantaranya yakni pada Sub Bab 3.2.1 Perancangan Modul Gangguan, sedangkan Krisna Sadewa pada perancangan Panel *Box* pada Sub Bab 3.2.2 Perancangan Panel *Box*.

Perancangan *hardware* dalam pembuatan alat yang digunakan oleh Iqbal Asad Kutsuro yakni pada Sub Bab 3.3 Perancangan *Hardware*.

Perancangan *software* dalam pembuatan alat yang digunakan oleh Krisna Sadewa yakni pada Sub Bab 3.4 Perancangan *Software*.

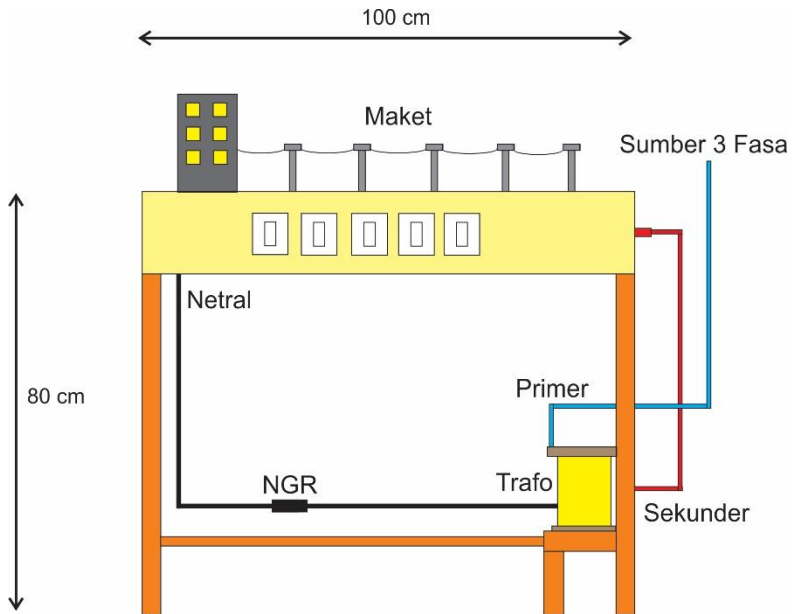
3.2.1 Perancangan Modul Gangguan

Pada prototipe modul gangguan ini yang dikerjakan oleh Iqbal Asad Kutsuro meliputi Sub Bab 3.2.1.1 Trafo *Step Down* 380V/12V, Sub Bab 3.2.1.2 Penentuan Besar Tahanan, Sub Bab 3.2.1.3 Penentuan Nilai NGR. Modul ini didesain mendekati sistem jaringan distribusi listrik dimana tetap menggunakan jaringan 3 fasa R, S, T, dan juga dilengkapi dengan jaringan netral atau grounding seperti Gambar 3.3. Modul gangguan ini merupakan representatif dari jaringan distribusi sistem tenaga listrik 20 kV yang dimulai dari trafo GI ke saluran penyulang hingga sampai ke beban atau pelanggan tegangan menengah. Nantinya panel pendeteksi gangguan akan di tempatkan pada posisi pangkal penyulang pada gardu induk.



Gambar 3.3 Skema Rangkaian Modul Gangguan

Pada perancangan modul simulasi gangguan ini dibuat dari kotak kayu yang memiliki dimensi 100 cm x 60 cm x 15 cm dan terdiri dari 5 buah sakelar gangguan (1 untuk tiap fasa dan 2 untuk antar fasa), 12 *resistor* dengan nilai 0,33 *Ohm* (4 untuk tiap fasa), MCB 6 *Ampere* 3 fasa, dan 3 buah lampu 15 *Watt* sebagai indikator pada Gambar 3.4. Untuk tiap *resistor* dengan nilai 0,33 *Ohm* mewakili nilai tahanan tiap 1 km pada konduktor AAAC pada SUTM. Terdapat 3 fasa R,S,T yang tiap fasanya terdapat 4 buah *resistor* 0,33 *Ohm* yang diseri. Selain itu pada modul gangguan ini juga dipasang lampu di masing-masing *resistor* untuk menggambarkan kondisi adanya beban pada setiap 1 km. untuk mempercantik modul gangguan juga akan dibuatkan maket yang menggambarkan kondisi seperti pada kondisi nyata sehingga diharapkan modul gangguan ini dapat memberi kesan yang bagus saat digunakan. Perancangan bangun modul gangguan dapat dilihat pada Gambar 3.4.



Gambar 3.4 Skema Prototipe Modul Gangguan

3.2.1.1 Trafo *Step Down* 380V / 12V

Pada modul simulasi gangguan ini digunakan Trafo *Step Down* 380/12 Volt sebagai sumber tegangan dan arus dari modul gangguan. Trafo *Step Down* merupakan suatu alat yang berhubungan dengan perangkat elektronik sebagai alat yang dapat menurunkan arus atau tegangan listrik. Pada Trafo *Step Down* ini memiliki jumlah kumparan sekunder lebih sedikit dibandingkan dengan jumlah kumparan primer. Hal ini dikarenakan dengan sedikitnya kumparan yang melilit medan magnet, maka arus yang dihasilkan akan semakin kecil, hal inilah yang menyebabkan mengapa jumlah kumparan sekunder lebih sedikit daripada kumparan primer pada trafo jenis ini. Untuk menentukan batasan arus *Short Circuit* maka harus mencari arus maksimal yang dapat ditahan oleh trafo dengan Persamaan 3.1.

$$I_{\max \text{ travo}} = \frac{kVa}{\sqrt{3.12}} = \frac{1000}{\sqrt{3.12}} = 48,12 \text{ A} \dots \dots \dots (3.1)$$

Penggunaan trafo *Step Down* adalah sebagai permodelan dari gardu induk yang menggunakan Trafo *Step Down* 70/20 kV di keadaan nyata. Dengan permodelan ini diharapkan nanti dapat menggantikan fungsi sumber tiga fasa yang digunakan karena pada modul gangguan ini akan sering dilakukan pengujian hubung singkat, maka dari itu untuk menghindari *trip* dari saluran tiga fasa langsung. Untuk model trafo dapat dilihat pada Gambar 3.5.



Gambar 3.5 Trafo *Step Down* 380/12 Volt

3.2.1.2 Penentuan Besaran Tahanan

Proses simulasi gangguan dilakukan dengan menentukan besaran tahanan dimana mewakili tahanan sepanjang konduktor SUTM dimana konduktor yang digunakan menggunakan tipe AAAC tanpa isolasi, berdasarkan SPLN (Standar Perusahaan Umum Listrik Negara) 64 : tahun 1995 tentang Impedansi Kawat Penghantar adapun spesifikasinya tertera pada Tabel 3.1.

Tabel 3.1 Tahanan dan Reaktansi Kawat Penghantar

Luas Penampang (mm ²)	Jari-Jari (mm ²)	Impedansi Urutan Positif (<i>Ohm/KM</i>)
16	2,2563	2,0161 + j 0,4036
25	2,8203	1,2903 + j 0,3895
35	3,3371	0,9217 + j 0,3790
50	3,9886	0,6452 + j 0,3678
70	4,7193	0,4608 + j 0,3572
95	5,4979	0,3096 + j 0,3449
150	5,2365	0,2162 + j 0,3305

Berdasarkan data pada Tabel 3.1 resistansi pada kabel AAAC yang sesuai dengan perancangan tugas akhir ini berdasarkan data penyulang Mayjend Sungkono yang menggunakan kabel AAAC dengan luas penampang 150 mm² yakni sebesar 0,2162 + j 0,3305 dimana sudah dalam bentuk impedansi urutan positif. Sehingga didapatkan impedansi saluran dengan menggunakan Persamaan 3.2 dimana R adalah resistansi dan X adalah reaktansi

$$Z = R + jX \dots\dots\dots (3.2)$$

Karena simulasi gangguan hanya berjarak 4 KM dimana tidak lebih dari 50 Mil maka jaringan dapat di kalsifikasikan menjadi saluran pendek sehingga hanya menggunakan R (Resistansi) dan X (Reaktansi) hanya diwakii oleh L (Induktansi).

$$Z = 0,2162 + j0,3305 \dots\dots\dots (3.3)$$

$$Z = \sqrt{0,2162^2 + 0,3305^2}$$

$$Z = 0,39 \text{ Ohm / km}$$

Dengan nilai impedansi sebesar 0,39 *Ohm/km* maka dengan mensubstitusikan Persamaan 3.2 hukum *Ohm* dengan Persamaan 3.3. Sehingga jarak gangguan dapat dicari dengan Persamaan 3.4.

$$Z = \frac{V_f}{I_f} \dots\dots\dots (3.4)$$

$$\text{Jarak} = Z \div 0,39 \dots\dots\dots (3.5)$$

$$\text{Jarak} = \frac{V_f}{I_f} \div 0,39 \dots\dots\dots (3.6)$$

Sehingga rumus untuk menentukan jarak didapat dengan menggunakan Persamaan 3.5 dan Persamaan 3.6.

3.2.1.3 Penentuan Nilai NGR (*Neutral Grounding Resistance*)

Pada kondisi nyata untuk sistem Jawa Timur yang menggunakan pentanahan dengan tahanan tinggi maka nilai tahanan yang digunakan adalah 500 *Ohm* dengan arus gangguan hubung singkat satu fasa ke tanah maksimal 25A jika menggunakan saluran udara. Untuk dapat membatasi arus *Short Circuit* sebesar 25 A seperti maka diperlukan perhitungan nilai NGR. Dengan Batasan arus 25 A maka nilai *resistor* yang digunakan didapat dengan menggunakan Persamaan 3.7.

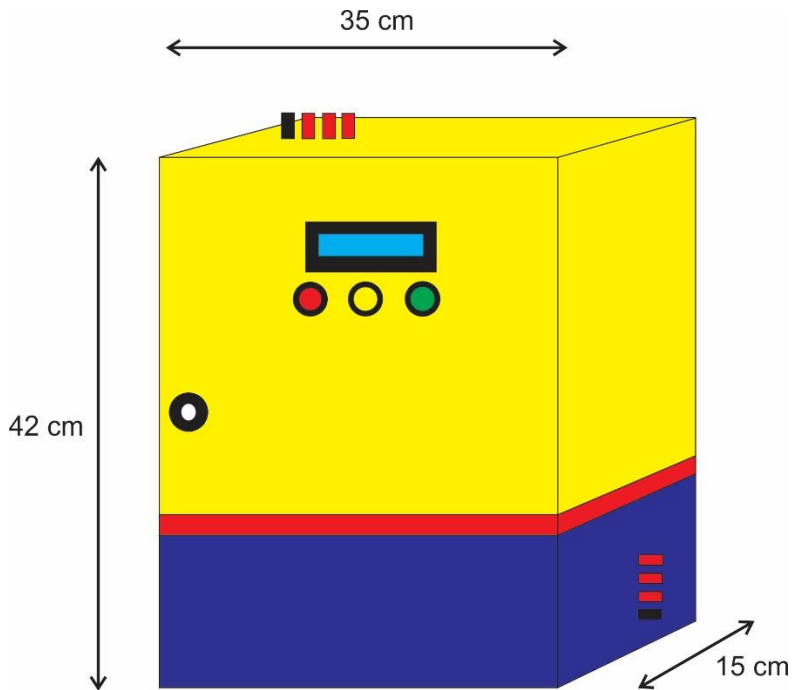
$$N_{gr} = \frac{V_{L-N}}{\frac{I_{sc}}{6,92}} \dots\dots\dots(3.7)$$

$$N_{gr} = \frac{6,92}{25}$$

$$N_{gr} = 0,27 \text{ } Ohm$$

3.2.2 Perancangan Panel *Box*

Pada perancangan Panel *Box* dibuat dari kotak berbahan kayu yang memiliki ukuran 35 cm x 15 cm x 42 cm. Pada kotak panel terdapat 3 buah lampu indikator dan layar lcd (16 x 2) pada Gambar 3.6. Ketiga buah lampu indikator digunakan untuk menandakan fasa mana yang mengalami gangguan, dan LCD akan menampilkan besar arus gangguan, fasa yang mengalami gangguan dan jarak gangguan.

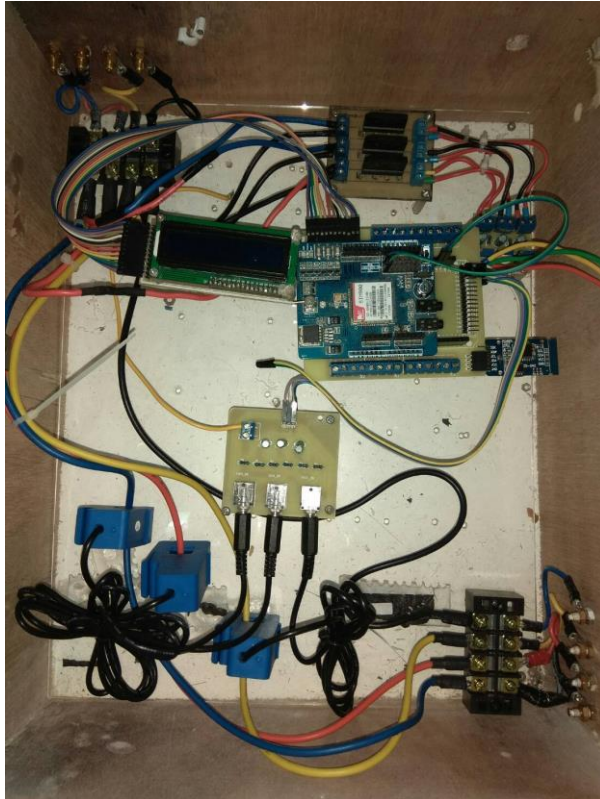


Gambar 3.6 Panel Box

3.3 Perancangan Hardware

Pada perancangan *hardware* pada tugas akhir ini yang dikerjakan oleh Krisna Sadewa meliputi Sub Bab 3.3.1 LCD (*Liquid Crystal Display*), Sub Bab 3.3.2 Perancangan *Power Supply*, Sub Bab 3.3.3 RTC (*Real Time Clock*), Sub Bab 3.3.4 Perancangan SIM900, Sub Bab 3.3.5 Sensor Arus, dan Sub Bab Sensor Tegangan yang terdiri dari rangkaian sensor arus YHDC 30A/1V, Perencanaan *Power Supply*, perancangan rangkaian sensor tegangan, modul RTC, rangkaian

Arduino Mega, rangkaian LCD, modem GSM SIM900. Tampilan perencanaan *hardware* terdapat pada Gambar 3.7.

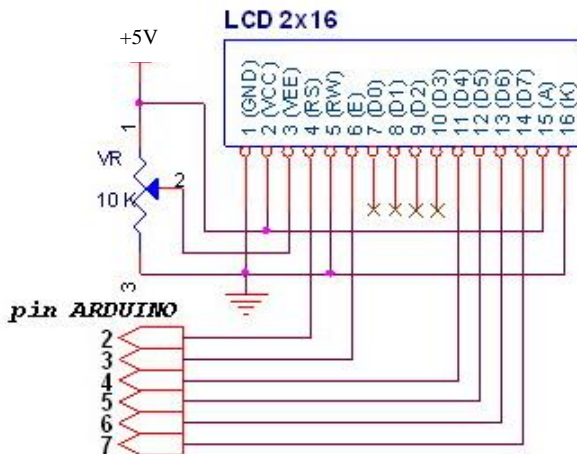


Gambar 3.7 Isi Panel *Box*

3.3.1 LCD (*Liquid Crystal Display*)

LCD berfungsi menampilkan karakter yang telah diproses oleh kontroler, yang bertujuan menginformasikan data yang sedang berlangsung atau telah diproses dalam bentuk karakter. LCD yang digunakan pada perancangan alat ini menggunakan LCD 16x2. Maksud dari 16x2 adalah mewakili 16 kolom dan 2 baris, berarti LCD ini dapat menampilkan karakter pada sepanjang 32 buah dengan masing-masing

kolom maksimal berisi 16 buah karakter. LCD ini membutuhkan *power supply* sebesar 5 Volt DC dan pin yang digunakan pada LCD pada alat ini yakni pin CONTR, RS, RW, E, D4, D5, D6, D7. Pada pin LCD yang diguakan di hubungkan dengan pin yang ada di Arduino yakni pin CONTR dihubungkan dengan potensio 10k *Ohm* untuk mengatur kecerahan pada karakter yang di tampilkan LCD pin RS, E, D4, D5, D6 dan D7 di hubungkan pada pin 7, 6, 5, 4, 3 dan 2 secara berurutan dan pin RW di hubungkan dengan *ground*. Rangkaian LCD ini dapat dilihat pada Gambar 3.8 berikut.



Gambar 3.8 Rangkaian LCD

Data yang akan di tampilkan pada LCD yakni kondisi Jaringan dan menampilkan tanggal, bulan dan waktu.

3.3.2 Perancangan Power Supply

Rangkaian *power supply* berfungsi sebagai catu daya Arduino Mega, sensor, dan modem SIM900. Oleh karena itu, pada tugas akhir ini digunakanlah *powerbank* sebagai *power supply* dengan tegangan sebesar

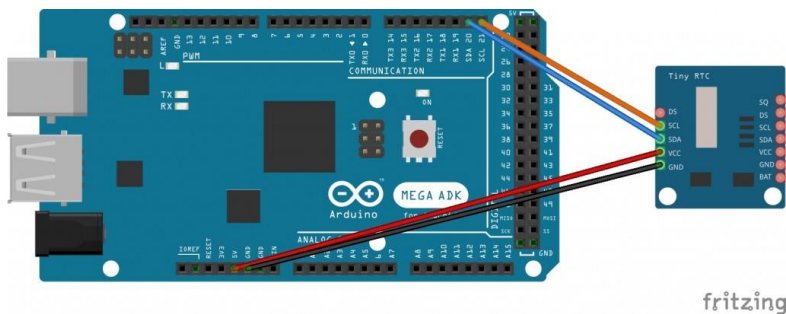
+5Volt. Rangkaian *power supply* dengan Arduino Mega untuk tugas akhir ini yang terlihat pada Gambar 3.9.



Gambar 3.9 Skema *Powerbank* ke Arduino Mega

3.3.3 RTC (*Real Time Clock*)

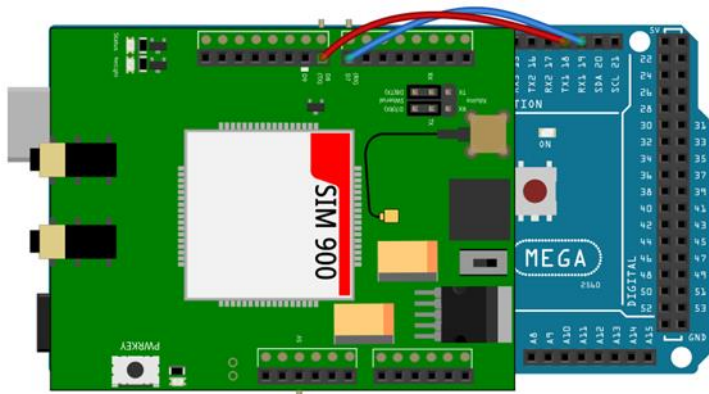
Rangkaian modul RTC dihubungkan pada Arduino Mega, SDA dan SCL pada pin modul RTC dihubungkan pada pin 20 dan pin 21 Arduino Mega, pin VCC pada RTC dihubungkan pada pin 5V Arduino Mega sedangkan pin GND di hubungkan dengan pin *Ground* pada Arduino Mega. Tipe RTC yang digunakan dalam tugas akhir ini adalah RTC DS3231. RTC DS3231 menyediakan pewaktu dalam detik, menit, jam, hari, tanggal, bulan dan tahun. Modul rangkaian RTC dapat dilihat pada Gambar 3.10.



Gambar 3.10 Skema Perancangan RTC

3.3.4 Perancangan SIM900

SIM900 berfungsi sebagai modem GSM atau modem GPRS. Pada SIM900 yang digunakan pada tugas akhir ini yakni sebagai modem GSM. SIM900 yang digunakan adalah “IComSat GSM / GPRS SIM900 Module Expansion Board Shield with Antenna For Arduino Mega” dimana modul ini berbentuk *shield* dimana dapat terintegrasi langsung dengan Arduino Mega 2560. SIM900 mengambil tegangan sumber sebesar 5 Volt DC dari Arduino Mega 2560. Pin 19 (RX1) dari Arduino Mega disambungkan dengan TX7 dari modul SIM900. Pin 18 (TX1) dari Arduino Mega tersambung dengan pin RX7 yang terdapat pada modul SIM900. Gambar dari SIM900 yang dihubungkan dengan Arduino Mega 2560 terdapat pada Gambar 3.11.



Gambar 3.11 Skema *Shield* SIM900 dengan Arduino Mega 2560

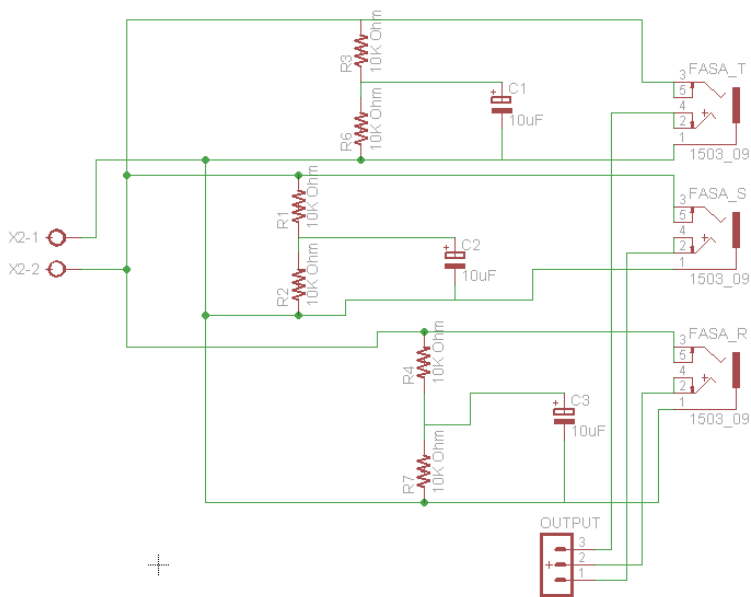
Pada perancangan SIM900 menggunakan pin serial 1 yakni pada pin 18, 19 yang digunakan untuk pengiriman data dan AT *command* dari Arduino ke *shield* SIM900, juga sebagai penerima data dari *shield* SIM900 ke Arduino.

3.3.5 Sensor Arus

Sensor arus merupakan sebuah alat yang digunakan untuk membaca nilai arus. Sensor yang digunakan pada perancangan alat ini yakni sensor arus YHDC 30 A/1 V. Pada perancangan simulasi gangguan, Karena arus yang di timbulkan oleh gangguan hubung singkat bernilai kurang dari 30 A maka menggunakan sensor arus dengan *rating* kerja 30 A pada Gambar 3.12 Sensor arus ini mempunyai pembacaan batas arus maksimum bernilai 30 A dan pada keluarannya menghasilkan tegangan bernilai maksimal 1 V DC oleh karena itu *output* dari sensor arus bisa langsung masuk dan terbaca di mikrokontroler dengan menggunakan rangkaian pengkondisi sinyal pada Gambar 3.13 yang berfungsi untuk menstabilkan *ripple* yang ada.



Gambar 3.12 Sensor Arus YHDC

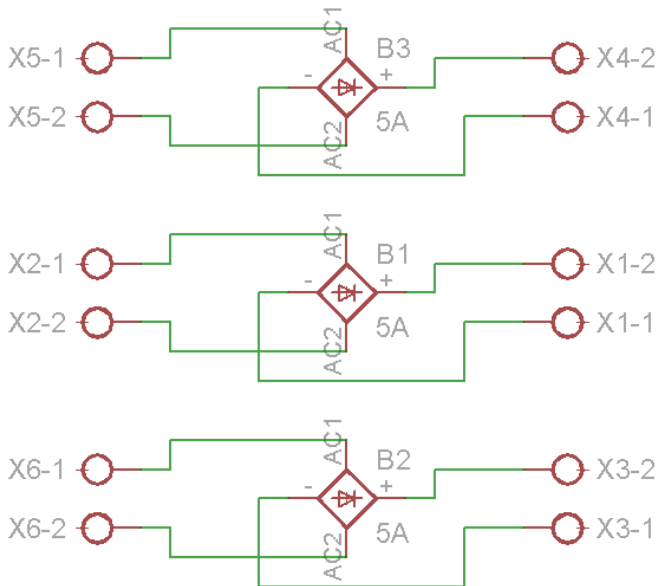


Gambar 3.13 Rangkaian Pengkondisi Sinyal

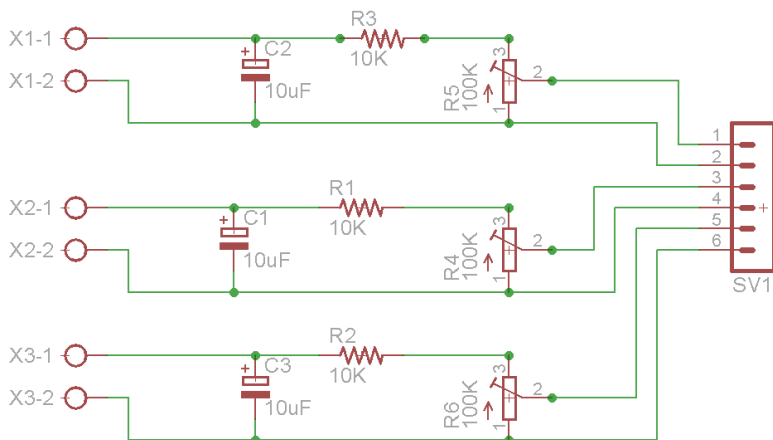
Pengkondisi sinyal pada sensor arus ini menggunakan rangkaian dengan kombinasi dua *resistor* 10 K pada R1 dan R2, R3 dan R6, dan R4 dan R7 lalu kapasitor 10 μ F pada C1, C2, dan C3. Sensor arus ini memungkinkan pembacaan lonjakan arus yang terjadi akibat hubung singkat. Perubahan yang akan dibaca yakni titik puncak arus yang terjadi sesaat saat terjadi hubung singkat.

3.3.6 Sensor Tegangan

Sensor tegangan merupakan alat yang digunakan untuk membaca nilai besaran tegangan yang melalui suatu peralatan listrik. Pada perancangan alat ini sensor tegangan menggunakan rangkaian pembagi tegangan dan rangkaian penyearah dengan skema perancangan pada Gambar 3.14 dan Gambar 3.15.



Gambar 3.14 Skema Rangkaian *Rectifier*



Gambar 3.15 Skema Rangkaian Pembagi Tegangan

Pada Gambar 3.15 menggunakan kapasitor 10 μ F 50 Volt pada C1, C2, dan C3. Resistor 10K ¼ Watt pada R1, R2, R3, dan juga *multiturn* 100 K pada R4, R5, R6 sebagai komponen pengatur nilai resistansi. Fungsi *multiturn* 10K ini sebagai alat pengatur tegangan yang telah disearahkan oleh *rectifier* sisir 5 A pada B1, B2, dan B3 yang terletak sebelum rangkaian pembagi tegangan.

3.4 Perancangan Software

Perancangan *software* di tugas akhir ini dikerjakan oleh Krisna Sadewa yang meliputi Sub Bab 3.4.1 Pemrograman Sistem Arduino IDE, Sub Bab 3.4.2 Pemrograman pada *APP Inventor*, dan Sub Bab 3.4.3 Pemrograman pada *Delphi* yang terbagi dari tiga bagian yakni perancangan *software* pada Arduino IDE, perancangan *software* dengan *App Inventor* yang akan menjadi tampilan pada Android, dan perancangan *software* dengan *Delphi* sebagai *server*.

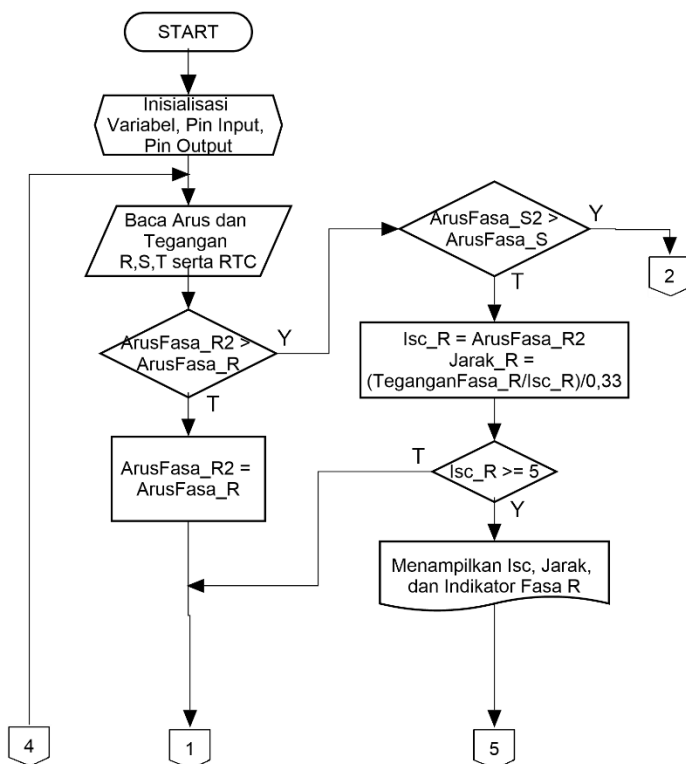
3.4.1 Pemrograman Sistem Arduino IDE

Pemrograman untuk *hardware* Arduino Mega menggunakan program Arduino IDE. *Flowchart* pemrograman mikrokontroler dapat dilihat pada Gambar 3.16, Gambar 3.17, dan Gambar 3.18. Berdasarkan *flowchart* yang terlihat pada Gambar 3.16 sampai 3.18 penjelasan algoritma pada program Arduino Mega, yakni :

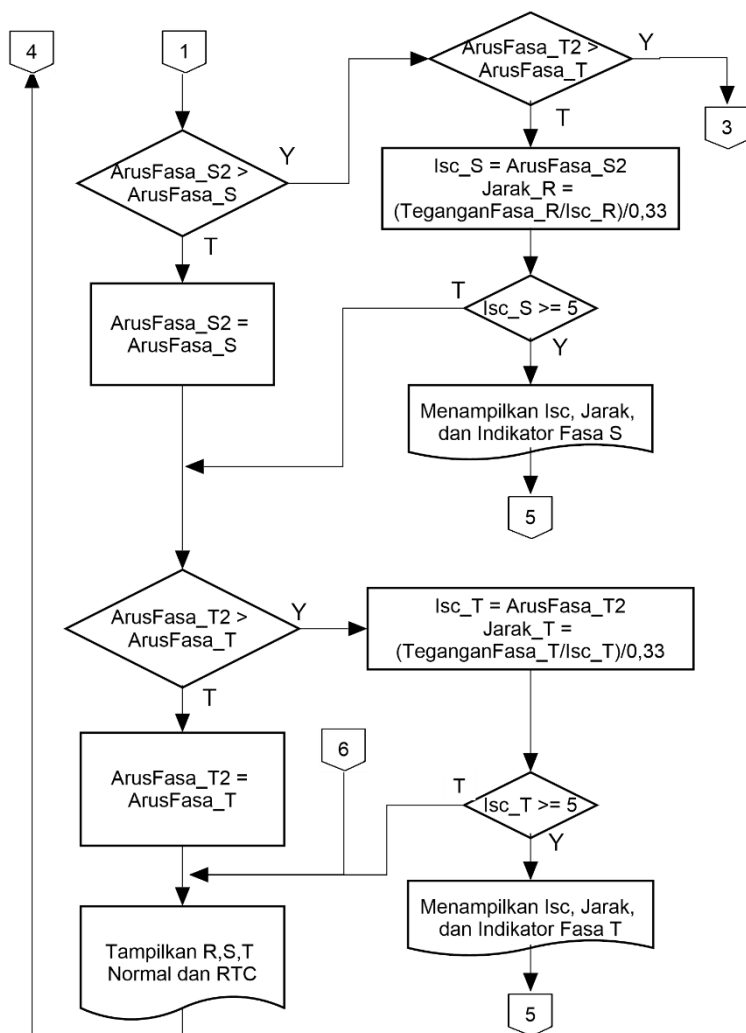
1. Arduino Mega aktif melakukan inisialisasi variabel data sensor arus dan sensor tegangan jalur R, S, T.
2. Pembacaan nilai arus dan nilai tegangan jalur R, S dan T
3. Jika ArusFasa_R2 (arus fasa R terbaca sebelumnya) lebih besar dari ArusFasa_R (arus fasa R terbaca sekarang) dan jika ArusFasa_S2 (arus fasa S terbaca sebelumnya) lebih besar dari ArusFasa_S (arus fasa S terbaca sekarang) maka I_{sc_RS} (arus hubung singkat dua fasa R dan S) = $(ArusFasa_R2 + ArusFasa_S2)/2$, TeganganFasa_RS (Tegangan rata-rata fasa R dan S) = $(TeganganFasa_R + TeganganFasa_S)/2$, dan perhitungan (jarak gangguan dua fasa R dan fasa S) $jarak_RS = (TeganganFasa_RS / I_{sc_RS}) / 0,33$.
4. Jika I_{sc_RS} lebih besar sama dengan 5 maka indikator fasa R dan fasa S gangguan akan menyala, LCD akan menampilkan fasa yang terganggu, jarak gangguan dan besar arus hubung singkat.

5. Setelah itu mengirimkan data gangguan meliputi jarak, Isc dan waktu terjadi gangguan melalui SMS.
6. Jika ArusFasa_R2 (arus fasa R terbaca sebelumnya) lebih besar dari ArusFasa_R (arus fasa R terbaca sekarang) dan jika ArusFasa_S2 (arus fasa S terbaca sebelumnya) tidak lebih besar dari ArusFasa_S (arus fasa S terbaca sekarang) maka Isc_R (arus hubung singkat fasa R) = ArusFasa_R2 dan perhitungan (jarak gangguan fasa R) $jarak_R = (TeganganFasa_R / Isc_R) / 0,33$.
7. Jika Isc_R lebih besar sama dengan 5 maka indikator fasa R gangguan akan menyala, LCD akan menampilkan fasa yang terganggu, jarak gangguan dan besar arus hubung singkat.
8. Setelah itu mengirimkan data gangguan meliputi jarak, Isc dan waktu terjadi gangguan melalui SMS.
9. Ketika arus fasa R terbaca sebelumnya tidak lebih besar dari arus fasa R terbaca sekarang maka arus fasa R terbaca sebelumnya = arus terbaca R terbaca sekarang.
10. Jika ArusFasa_S2 (arus fasa S terbaca sebelumnya) lebih besar dari ArusFasa_S (arus fasa S terbaca sekarang) dan jika ArusFasa_T2 (arus fasa T terbaca sebelumnya) lebih besar dari ArusFasa_T (arus fasa T terbaca sekarang) maka Isc_ST (arus hubung singkat dua fasa S dan T) = $(ArusFasa_S2 + ArusFasa_T2)/2$, TeganganFasa_ST (Tegangan rata-rata fasa S dan T) = $(TeganganFasa_S + TeganganFasa_T)/2$, dan perhitungan (jarak gangguan dua fasa S dan fasa T) $jarak_ST = (TeganganFasa_ST / Isc_ST) / 0,33$.
11. Jika Isc_ST lebih besar sama dengan 5 maka indikator fasa S dan fasa T gangguan akan menyala, LCD akan menampilkan fasa yang terganggu, jarak gangguan dan besar arus hubung singkat.
12. Setelah itu mengirimkan data gangguan meliputi jarak, Isc dan waktu terjadi gangguan melalui SMS.
13. Jika ArusFasa_S2 (arus fasa S terbaca sebelumnya) lebih besar dari ArusFasa_S (arus fasa S terbaca sekarang) dan jika ArusFasa_T2 (arus fasa T terbaca sebelumnya) tidak lebih besar dari ArusFasa_T (arus fasa T terbaca sekarang) maka Isc_S (arus hubung singkat fasa S) = ArusFasa_S2 dan perhitungan (jarak gangguan fasa S) $jarak_S = (TeganganFasa_S / Isc_S) / 0,33$.
14. Jika Isc_S lebih besar sama dengan 5 maka indikator fasa S gangguan akan menyala, LCD akan menampilkan fasa yang terganggu, jarak gangguan dan besar arus hubung singkat.

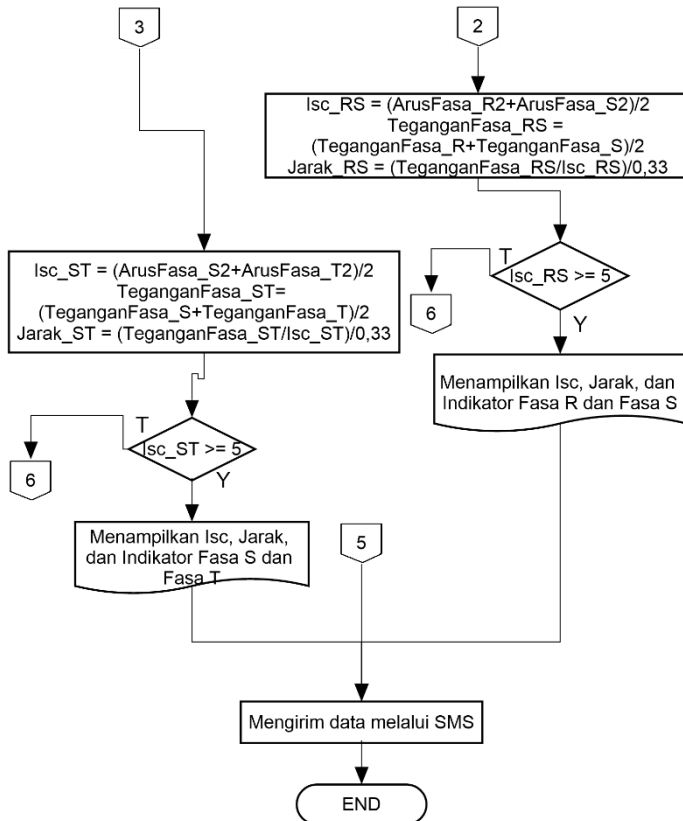
15. Setelah itu mengirimkan data gangguan meliputi jarak, Isc dan waktu terjadi gangguan melalui SMS.
16. Ketika arus fasa S terbaca sebelumnya tidak lebih besar dari arus fasa S terbaca sekarang maka arus fasa S terbaca sebelumnya = arus terbaca S terbaca sekarang.
17. Jika ArusFasa_T2 (arus fasa T terbaca sebelumnya) lebih besar dari ArusFasa_T (arus fasa T terbaca sekarang) maka Isc_T (arus hubung singkat fasa T) = ArusFasa_T2 dan perhitungan (jarak gangguan fasa T) $\text{jarak}_T = (\text{TeganganFasa}_T / \text{Isc}_T) / 0,33$.
18. Ketika arus fasa T terbaca sebelumnya tidak lebih besar dari arus fasa T terbaca sekarang maka arus fasa T terbaca sebelumnya = arus terbaca T terbaca sekarang.



Gambar 3.16 Flowchart 1 Program Arduino



Gambar 3.17 Flowchart 2 Program Arduino



Gambar 3.18 Flowchart 3 Program Arduino

3.4.2 Pemrograman pada App Inventor

Pada pemrograman App Inventor terbagi menjadi 3 Screen yang akan menjadi tampilan Android ketika menerima SMS yang dikirimkan oleh hardware.

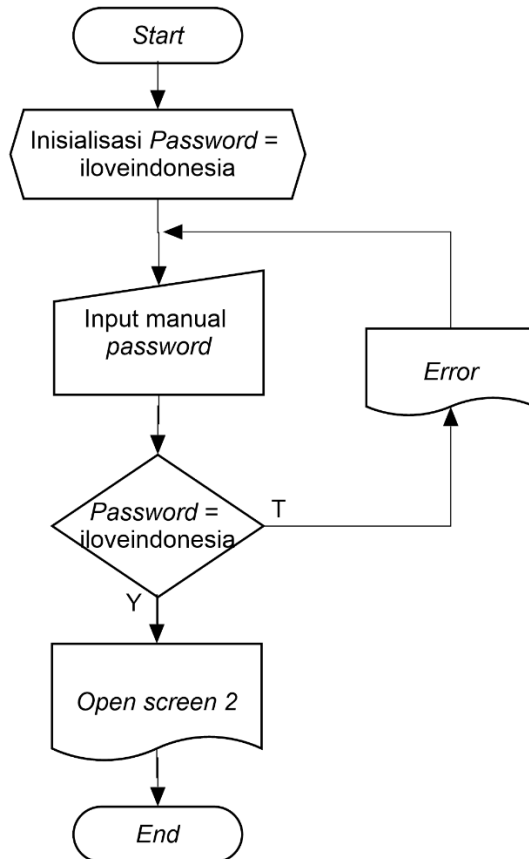
1. Tampilan screen 1 pada aplikasi

Tampilan pada screen 1 merupakan tampilan halaman utama ketika aplikasi pertama kali dibuka di tunjukkan pada Gambar 3.19. Pada screen 1 ini terdapat tombol access untuk mengakses aplikasi selain itu juga dilengkapi pengaman berupa

password untuk bisa mengakses aplikasi ini. *Flowchart* pemrograman pada *screen 1* terlihat pada Gambar 3.20.



Gambar 3.19 Tampilan *Screen 1*

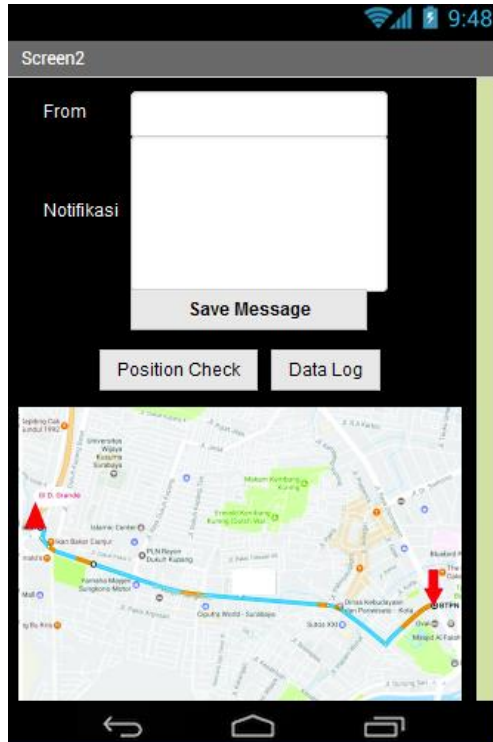


Gambar 3.20 Flowchart Screen 1

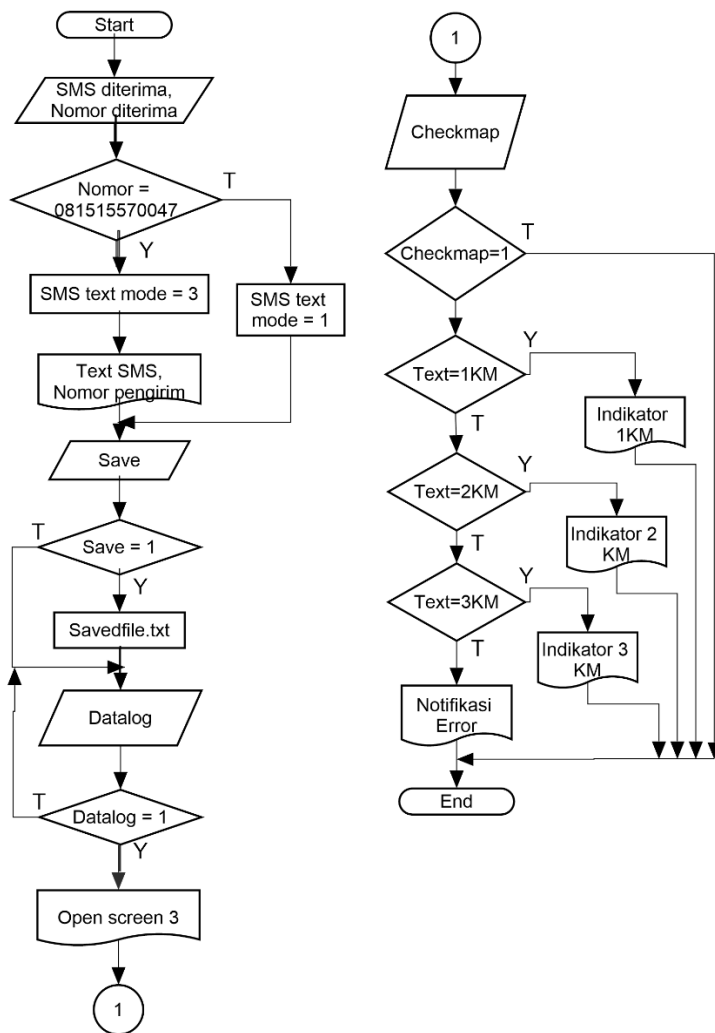
2. Tampilan *screen 2* pada aplikasi

Pada tampilan *screen 2* terdapat *textbox* yang akan menampilkan pesan yang telah diterima dan nomor pengirim terlihat pada Gambar 3.21. Pada halaman ini juga terdapat beberapa tombol diantaranya *save message* untuk menyimpan pesan, *check on map* untuk melihat lokasi gangguan pada peta lokasi penyulang Meyjend Sungkono dan tombol *Data Log* untuk melihat seluruh SMS yang pernah diterima sebelumnya.

Flowchart pemrograman *Block* pada *screen 2* terlihat pada Gambar 3.22.



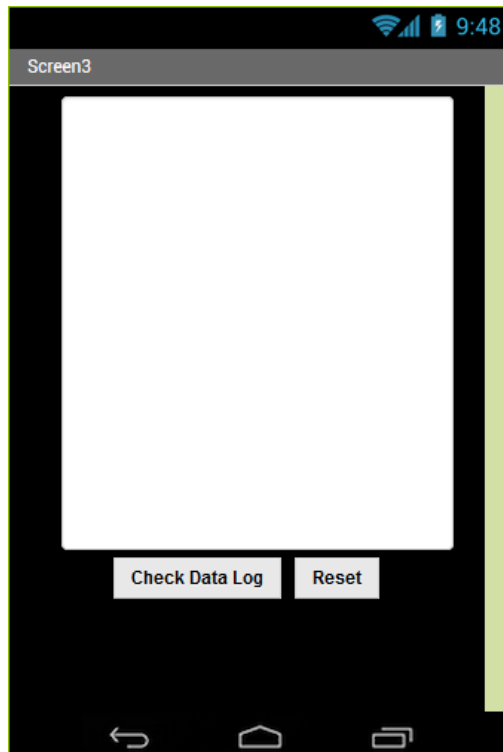
Gambar 3.21 Tampilan *Screen 2*



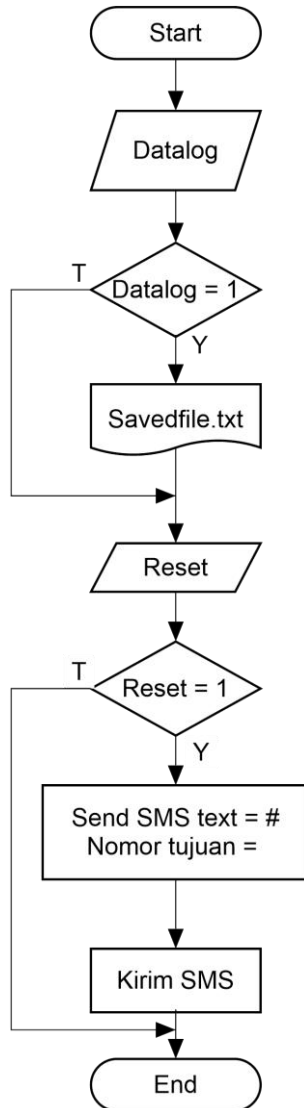
Gambar 3.22 Flowchart Screen 2

3. Tampilan *screen 3* pada aplikasi

Pada tampilan *screen 3* merupakan *screen* yang digunakan untuk mengakses *data log* dari SMS gangguan yang telah diterima sebelumnya, dengan menekan tombol *check data log*. Yang terakhir adalah tombol *reset* untuk melakukan *reset* mikrokontroler secara *remote* melalui Android. Tampilan *screen 3* dapat dilihat pada Gambar 3.23 pemrograman *block screen 3* ditunjukkan pada Gambar 3.24.



Gambar 3.23 Tampilan *Screen 3*



Gambar 3.24 *Flowchart Screen 3*

3.4.3 Pemrograman pada Delphi

Pada tugas akhir ini perancangan aplikasi *server* menggunakan *software Delphi* untuk membuat tampilan HMI (*Human Machine Interface*). Desain tampilan HMI dapat dilihat pada Gambar 3.25 Serta *Flowchart* untuk pemrograman pada *Delphi7* dapat dilihat pada Gambar 3.26.

Pelaporan Gangguan
Jaringan Tegangan Menengah 20 KV

Fasa Terganggu

R S T

Records

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangg ^

Rincian Gangguan

Tanggal

Waktu

Jenis Gangguan

Fasa Yang Terganggu

Jarak Gangguan

Connection

Set Port Open

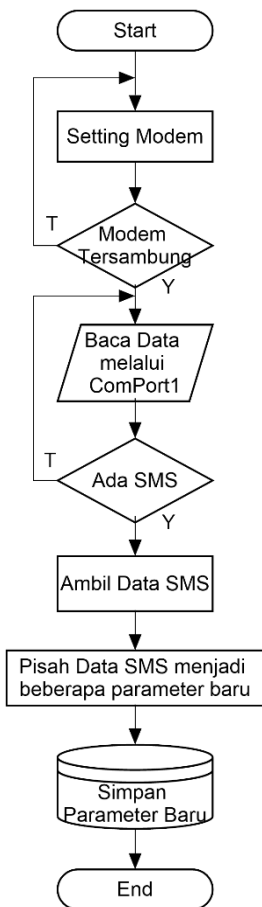
Clear Close App

Gambar 3.25 Tampilan HMI (*Human Machine Interface*) pada *Delphi*

Berdasarkan *flowchart* pada Gambar 3.26 dapat dijabarkan algoritmanya sebagai berikut:

1. Ketika program di *Run* yang dilakukan oleh *user* pertama kali yakni melakukan setting modem.
2. *Delphi* akan melakukan pengecekan , apabila modem telah tersambung maka akan dilakukan pembacaan melalui *ComPort1*. Jika belum tersambung maka akan dilakukan setting ulang modem.
3. Membaca data melalui *ComPort1*, kemudian dilakukan pengecekan apakah ada SMS masuk?, Jika ada maka ambil data

- SMS. Jika tidak maka lakukan pengecekan lagi ada SMS atau tidak.
4. Ketika data SMS sudah diambil maka lakukan pemisahan data SMS menjadi beberapa parameter baru.
 5. Setelah data SMS dipisah menjadi parameter baru simpan parameter baru tersebut pada *database*
 6. Program selesai.



Gambar 3.26 Flowchart Pemrograman HMI pada Delphi

-----*Halaman ini sengaja dikosongkan*-----

BAB IV

PENGUJIAN DAN ANALISA DATA

Pada bab ini akan membahas mengenai hasil pengujian dan Analisa atas penyusunan alat untuk “Desain Prototipe Sistem Deteksi Gangguan Impedansi Tinggi Pada SUTM 20 kV dengan Sistem Pelaporan Gangguan Menggunakan Komunikasi Data Modul GSM “. Data pengujian sangat diperlukan untuk implementasi dalam dunia nyata. Pengujian sistem ini untuk mengetahui apakah alat yang sudah dibuat bisa berjalan dengan baik dan dapat diambil analisa dari sistem kerja alat tersebut seperti pada Gambar 4.1.



Gambar 4.1 Pengujian Keseluruhan Sistem

Pengujian merupakan salah satu langkah yang harus dilakukan untuk mengetahui apakah sistem yang telah dibuat sesuai dengan yang direncanakan. Kesesuaian sistem dengan perencanaan dapat dilihat dari hasil-hasil yang dicapai pada pengujian sistem. Pengujian juga bertujuan untuk mengetahui kelebihan dan kekurangan dari sistem yang telah dibuat. Hasil pengujian tersebut akan dianalisa untuk mengetahui penyebab terjadinya kekurangan atau kesalahan dalam sistem. Pengujian tersebut meliputi :

1. Pengujian Arduino Mega 2560
2. Pengujian *Power Supply*
3. Pengujian *Real Time Clock (RTC)*
4. Pengujian LCD (*Liquid Crystal Display*)

5. Pengujian CT Sensor YHDC ST-013
6. Pengujian Sensor Tegangan
7. Pengujian Komunikasi SIM 900
8. Pengujian *Short Circuit* dan *Open Circuit Transformator*
9. Pengujian Arus Hubung Singkat
10. Pengujian Modul Gangguan
11. Pengujian Keseluruhan

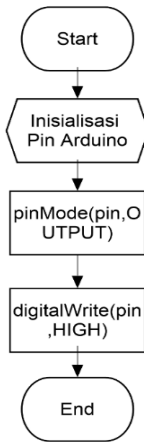
Dari pengujian yang telah dibuat tersebut, selanjutnya Krisna Sadewa akan melakukan pengujian dan analisa dari berbagai komponen sistem diantaranya pada Sub Bab 4.1 Pengujian Arduino Mega 2560 dan Sub Bab 4.7 Pengujian Komunikasi SIM900.

Dari *hardware* yang telah dibuat tersebut, selanjutnya Iqbal Asad Kutsuro akan melakukan pengujian dan analisa dari berbagai komponen sistem diantaranya pada Sub Bab 4.8 Pengujian *Short Circuit* dan *Open Circuit*, pada Sub Bab 4.9 Pengujian Arus Hubung Singkat dan Sub Bab 4.10 Pengujian Modul Gangguan.

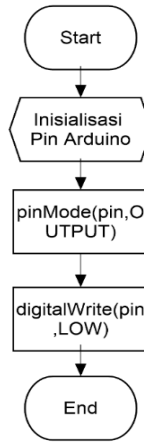
Setelah semuanya selesai nantinya Krisna Sadewa dan Iqbal Asad Kutsuro akan melakukan pengujian dan Analisa dari Sub Bab 4.11 Pengujian Keseluruhan.

4.1 Pengujian Arduino Mega 2560

Pengujian ini dilakukan terhadap *board* Arduino Mega 2560. Pengujian ini dilakukan agar dapat mengetahui kondisi arduino saat aktif, selain itu dapat diketahui apakah arduino dalam kondisi bagus atau tidak. Pengujian dilakukan pada pin yang digunakan pada sistem alat ini yakni pin A0 s/d A7, D0 s/d D7, D14 s/d D21 milik Arduino Mega dengan cara mengatur keluarannya pada logika 1 dan 0 dengan mengunggah program ke *board* Arduino. Gambar 4.2 (a) merupakan program untuk memberikan tegangan pada pin Arduino berlogika 1 atau sebesar 5 *Volt* sedangkan pada Gambar 4.2 (b) merupakan program untuk memberikan tegangan keluaran pada pin Arduino berlogika 0 atau 0 *Volt*. Kemudian memeriksa besar keluaran tegangan pada setiap pin dan pada pengujian ini seperti pada Gambar 4.3. Data yang didapat adalah pada Tabel 4.1.

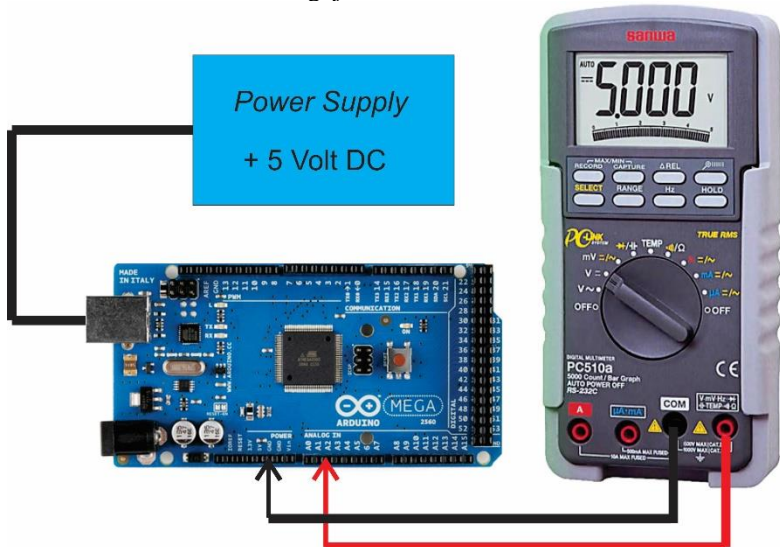


(a)



(b)

Gambar 4.2 Flowchart Pengujian Arduino



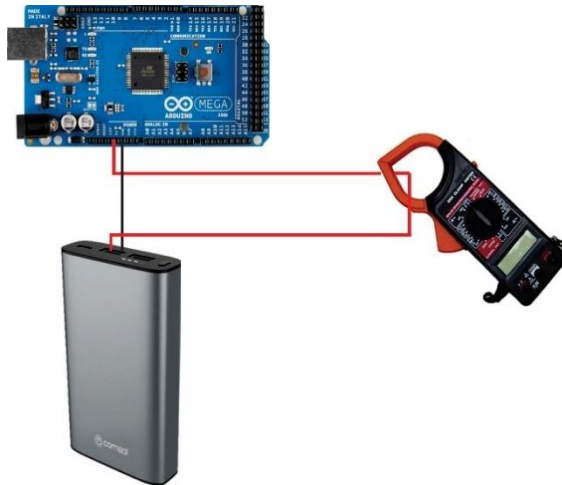
Gambar 4.3 Pengujian Arduino

Tabel 4.1 Hasil Pengujian Arduino Mega 2560

Pin	V Logika 1	V Logika 0
A0	4,99 Volt	0 Volt
A1	4,99 Volt	0 Volt
A2	4,99 Volt	0 Volt
A3	4,99 Volt	0 Volt
A4	4,99 Volt	0 Volt
A5	4,99 Volt	0 Volt
A6	4,99 Volt	0 Volt
A7	4,99 Volt	0 Volt
D0	4,99 Volt	0 Volt
D1	4,99 Volt	0 Volt
D2	4,99 Volt	0 Volt
D3	4,99 Volt	0 Volt
D4	4,99 Volt	0 Volt
D5	4,99 Volt	0 Volt
D6	4,99 Volt	0 Volt
D7	4,99 Volt	0 Volt
D14	4,99 Volt	0 Volt
D15	4,99 Volt	0 Volt
D16	4,99 Volt	0 Volt
D17	4,99 Volt	0 Volt
D18	4,99 Volt	0 Volt
D19	4,99 Volt	0 Volt
D20	4,99 Volt	0 Volt

4.2 Pengujian Power Supply

Dalam pengujian *power supply* ini menggunakan *powerbank* berkapasitas 12000 mAh dengan *output* DC 5 V/2.4 A. Pengujian pada *powerbank* ini dilakukan agar kita dapat mengetahui apakah *powerbank* ini bekerja dengan bagus dan tidak terdapat kerusakan. Selain itu pengujian ini dilakukan untuk memastikan apakah *powerbank* ini kompatibel dengan arduino. Prosedur pengujian yang di diambil yakni data tegangan dan arus saja. Pengujian arus dilakukan dengan menggunakan *clampmeter* ke salah satu kabel seperti Gambar 4.4 lalu untuk pengujian tegangan dilakukan dengna mengukur tegangan menggunakan *multimeter* pada Gambar 4.5.



Gambar 4.4 Skema Pengujian Arus *Power Supply*



Gambar 4.5 Skema Pengujian *Power Supply*

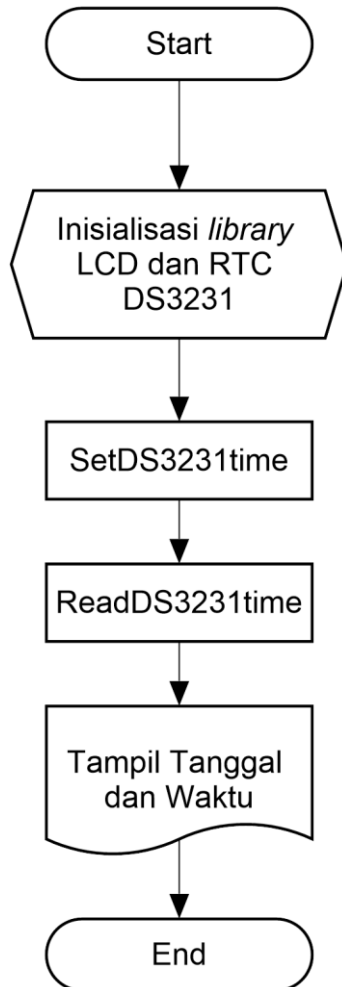
Dari data pada Tabel 4.2 bahwa tegangan keluaran *powerbank* 5,12 Volt. Nilai ini masih termasuk *range* kerja mikrokontroler atmega 2560 yakni 3,3 – 5,5 Volt. Sehingga tegangan *powerbank* masih dapat digunakan untuk memberikan *supply* ke rangkaian Arduino.

Tabel 4.2 Pengujian *Powerbank*

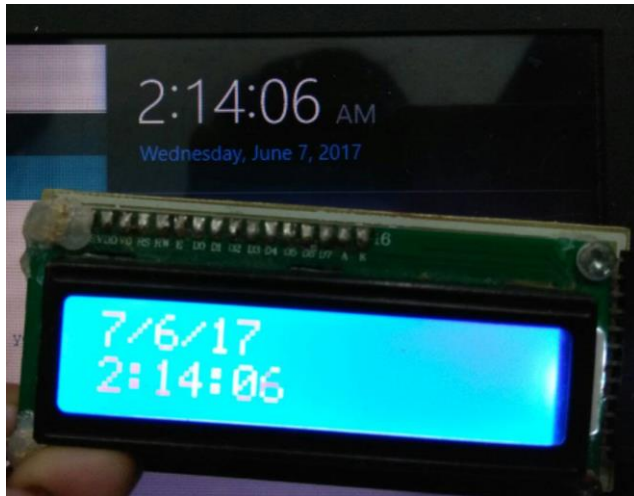
V out	I out
5,12 V	0,15 A

4.3 Pengujian *Real Time Clock (RTC)*

Pengujian pada RTC ini dilakukan agar dapat mengetahui berapakah *error* waktu yang dihasilkan oleh RTC karena RTC adalah merupakan modul yang mengolah data waktu seperti waktu asli tetapi harus disesuaikan terlebih dahulu menggunakan program. Pengujian ini dilakukan dengan menggunakan *flowchart* program yang dimasukkan ke arduino pada Gambar 4.6.



Gambar 4.6 *Flowchart* Pengujian RTC



Gambar 4.7 Pengujian RTC

Pada Tabel 4.3 ditunjukkan hasil pengujian RTC. Hasil pengujian menunjukkan waktu dan tanggal pada RTC telah sesuai dengan tanggal dan waktu pada PC pada Gambar 4.7 dalam 5 kali pengambilan data per detik dengan selisih rata-rata 0 detik, maka dapat disimpulkan bahwa RTC telah sesuai dan dapat digunakan pada sistem.

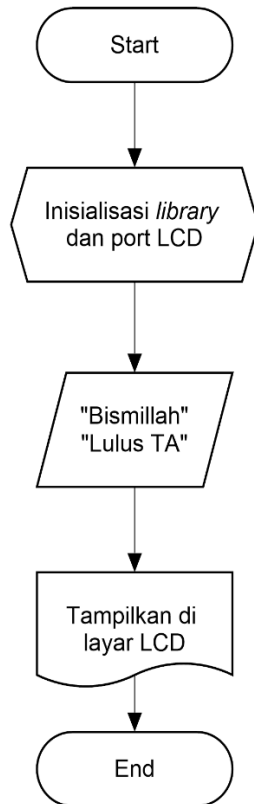
Tabel 4.3 Hasil Pengujian RTC

No	Waktu RTC	Waktu PC	Selisih
1	01 : 39 : 54	01 : 39 : 54	0 Detik
2	01 : 42 : 07	01 : 42 : 07	0 Detik
3	01 : 43 : 34	01 : 43 : 34	0 Detik
4	01 : 50 : 46	01 : 50 : 46	0 Detik
5	02 : 00 : 01	02 : 00 : 01	0 Detik

4.4 Pengujian *Liquid Crystal Display (LCD)*

Pengujian LCD ini dilakukan agar kita dapat mengetahui apakah LCD bekerja dalam kondisi bagus dan tidak ada kerusakan pada LCD. Pengujian LCD dilakukan dengan cara menambahkan program pada mikrokontroler Arduino untuk menampilkan karakter pada LCD. Pada pengujian LCD kali ini menggunakan LCD 16 x 2, pengujian kami lakukan dengan memberi program untuk menampilkan karakter “

BISMILLAH” pada baris pertama dan “LULUS TA” pada baris selanjutnya pada Arduino *flowchart* pemrograman pada Gambar 4.8 dan hasil tampilan pada layar LCD di tunjukkan pada Gambar 4.9.



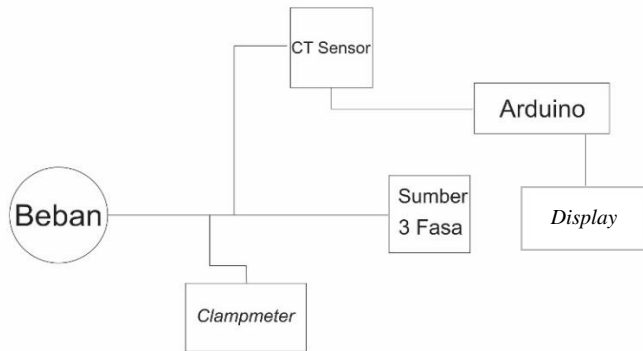
Gambar 4.8 *Flowchart* Pengujian LCD



Gambar 4.9 Hasil Tampilan LCD

4.5 Pengujian CT Sensor YHDC ST-013

Pada pengujian sensor arus ini, kami melakukan kalibrasi CT Sensor YHDC ST-013 untuk mendapatkan tingkat akurasi pembacaan arus sebaik mungkin. Pengujian dilakukan kepada 3 sensor arus yang kami gunakan yang nantinya akan mewakili fasa R, S, T dengan menggunakan variasi arus gangguan yang berubah ubah nilainya. Pengujian ini dilakukan dengan menguji CT sensor satu per satu dimana CT 1 mewakili fasa R, CT 2 mewakili fasa S, dan CT 3 mewakili fasa T. Pengujian ini dilakukan agar kita bisa mendapatkan nilai analog yang dibaca oleh arduino dari nilai tegangan *output* CT sensor yang sudah berupa tegangan yang muncul pada *display*. Kalibrasi yang dilakukan ini dilakukan satu per satu untuk masing-masing CT sensor. Pada pengukuran pertama digunakan Persamaan ICAL yang diambil dari *datasheet* CT Sensor YHDC ST-013. Skema pengujian CT sensor dapat dilihat pada Gambar 4.10.



Gambar 4.10 Skema Pengujian CT Sensor

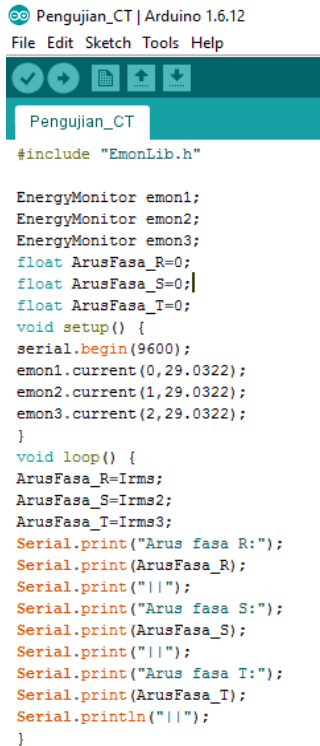
Untuk mendapatkan hasil yang sesuai maka perlu adanya inisialisasi penentuan ICAL atau Persamaan kalibrasi yang dapat dihitung dengan menggunakan Persamaan 4.1.

$$ICAL = \frac{\frac{\text{Jumlah Lilitan Primer}}{\text{Jumlah Lilitan Sekunder}}}{\text{Resistansi Burden}} \dots\dots\dots(4.1)$$

$$ICAL = \frac{1800}{62}$$

$$ICAL = 29,0322$$

Dengan memasukan nilai ICAL sebagai persamaan awal kemudian kita masukkan kedalam program pengujian CT sensor pada Gambar 4.11 maka akan didapatkan nilai pengukuran awal pada Tabel 4.4.

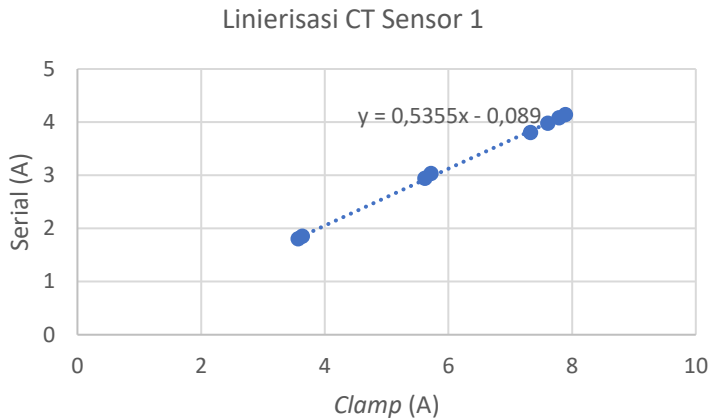


Gambar 4.11 Program Pengujian CT Sensor

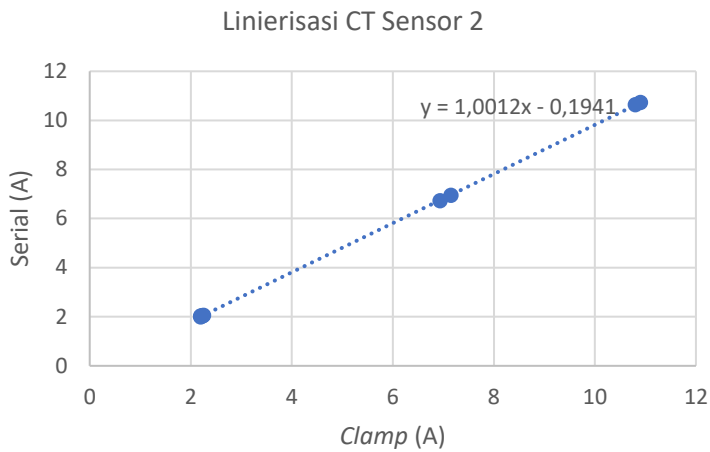
Tabel 4.4 Hasil *Sensing* Awal CT Sensor

No	CT 1		CT 2		CT 3	
	Clamp (A)	Serial (A)	Clamp (A)	Serial (A)	Clamp (A)	Serial (A)
1	3,64	1,85	2,25	2,06	2,25	2
2	7,89	4,14	10,8	10,63	10,8	10,5
3	7,61	3,98	2,19	2,03	2,19	1,96
4	5,72	3,03	6,93	6,73	6,93	6,76
5	5,62	2,94	2,19	1,99	2,19	1,77
6	3,57	1,8	10,9	10,73	10,9	10,7
7	7,79	4,08	2,23	2,04	2,23	2,01

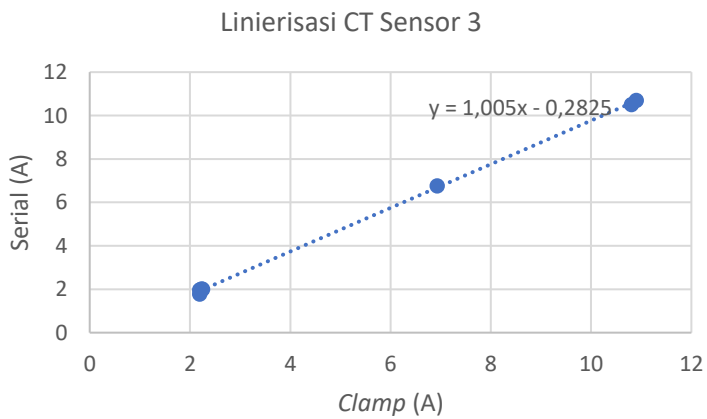
Dari Tabel 4.4 didapatkan data pengujian CT sensor yang masuk ke Arduino dapat dibuat suatu persamaan karakteristik linierisasi sensor arus yang akan digunakan pada pembacaan nilai arus pada Arduino yang sesuai dengan nilai arus pengukuran menggunakan persamaan *scatter* pada *Microsoft Excel*. Linierisasi CT 1 fasa R ditunjukkan pada Gambar 4.12, CT 2 fasa S ditunjukkan pada Gambar 4.13, dan linierisasi CT 3 fasa T ditunjukkan pada Gambar 4.14.



Gambar 4.12 Linierisasi CT Sensor 1



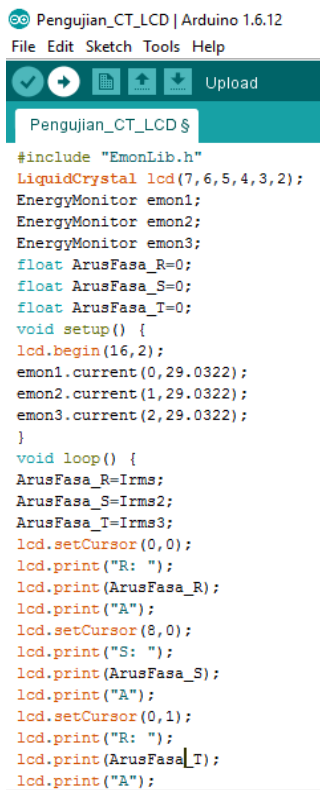
Gambar 4.13 Linierisasi CT Sensor 2



Gambar 4.14 Linierisasi CT Sensor 3

Setelah didapatkan Persamaan linearnya dari masing-masing fasa kemudian dilakukan pengujian dengan menggunakan Persamaan yang dimasukkan ke dalam program Arduino dan ditampilkan pada LCD pada Gambar 4.15. Hasil pengujian dengan Persamaan tersebut dilihat dengan mengukur arus hubung singkat yang berbeda nilainya. Perhitungan presentase kesalahan pengukuran sensor arus didapat dari Persamaan 4.2. Untuk hasil pengujian presentase *error* sensor arus ini secara lengkap tiap fasa R, S dan T dapat dilihat pada Tabel 4.5.

$$Error = \frac{Hasil\ alat\ ukur - Tampilan\ LCD}{Hasil\ alat\ ukur} \times 100\% \dots \dots \dots (4.2)$$



```

Pengujian_CT_LCD | Arduino 1.6.12
File Edit Sketch Tools Help

Upload

Pengujian_CT_LCD $
#include "EmonLib.h"
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
EnergyMonitor emon1;
EnergyMonitor emon2;
EnergyMonitor emon3;
float ArusFasa_R=0;
float ArusFasa_S=0;
float ArusFasa_T=0;
void setup() {
  lcd.begin(16, 2);
  emon1.current(0, 29.0322);
  emon2.current(1, 29.0322);
  emon3.current(2, 29.0322);
}
void loop() {
  ArusFasa_R=Irms;
  ArusFasa_S=Irms2;
  ArusFasa_T=Irms3;
  lcd.setCursor(0, 0);
  lcd.print("R: ");
  lcd.print(ArusFasa_R);
  lcd.print("A");
  lcd.setCursor(8, 0);
  lcd.print("S: ");
  lcd.print(ArusFasa_S);
  lcd.print("A");
  lcd.setCursor(0, 1);
  lcd.print("T: ");
  lcd.print(ArusFasa_T);
  lcd.print("A");
}

```

Gambar 4.15 Program Tampilan LCD Pengujian CT Sensor

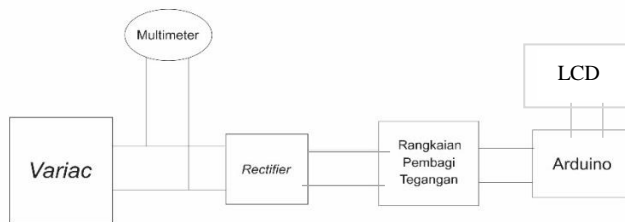
Tabel 4.5 Hasil Pengujian CT Sensor

No	Arus (A)	CT 1		CT 2		CT 3	
		LCD	Error (%)	LCD	Error (%)	LCD	Error (%)
1	2,23	2,26	1,35	2,25	0,90	2,19	1,79
2	10,9	11	0,92	11,2	2,75	10,5	3,67
3	7,33	7,29	0,55	7,4	0,95	7,3	0,41
4	2,24	2,2	1,79	2,27	1,34	2,21	1,34
5	7,25	7,23	0,28	7,21	0,55	7,20	0,69
6	2,23	2,19	1,79	2,2	1,35	2,2	1,35
7	11,2	11,4	1,79	11	1,79	10,9	2,68

Dari data Tabel 4.5 yang didapat dari pengujian, dapat dihitung rata-rata *error* pada sensor arus fasa R sebesar 1,8%, sensor arus fasa S sebesar 1,97%, dan sensor arus fasa T sebesar 1,55%. Perbedaan dari keluaran sensor tersebut dapat dikarenakan faktor internal maupun faktor eksternal. Faktor internal berasal dari karakteristik tiap komponen yang tidak semuanya memiliki standar yang baik. Faktor eksternal berasal dari tegangan sumber yang masuk tiap-tiap sensor tidak sama dan juga faktor dari kesalahan pembacaan alat ukur.

4.6 Pengujian Sensor Tegangan

Sensor tegangan digunakan untuk mengukur besar tegangan yang ada. Sensor tegangan yang dipakai adalah rangkaian pembagi tegangan. Sensor tegangan diuji dengan menggunakan *Variac* dan *multimeter*. Pengujian dilakukan dengan menyambungkan satu *rectifier* yang mewakili satu fasa dengan *variatic*, kemudian diukur masing-masing nilai tegangan yang keluar dan nilai ADC yang muncul pada LCD. Pengujian dilakukan satu per satu untuk mewakili tiap fasa dengan skema pada Gambar 4.16. Untuk hasil dari pengujian ketiga sensor tegangan pada fasa R diwakili oleh ADC 1, fasa S diwakili oleh ADC 2, dan fasa T diwakili oleh ADC 3 yang dimasukkan kedalam program pengujian sensor tegangan pada Gambar 4.17 kemudian hasil dari pengujian dapat dilihat pada Tabel 4.6.



Gambar 4.16 Skema Pengujian Sensor Tegangan

🔗 Pengujian_pt_adc | Arduino 1.6.12
File Edit Sketch Tools Help

```

Pengujian_pt_adc $
float TeganganFasa_R=0;
float TeganganFasa_S=0;
float TeganganFasa_T=0;
void setup() {
  serial.begin(9600);
}
void loop() {
  TeganganFasa_R=(analogRead(A7));
  TeganganFasa_S=(analogRead(A6));
  TeganganFasa_T=(analogRead(A5));
  Serial.print("ADC fasa R:");
  Serial.print(TeganganFasa_R);
  Serial.print("||");
  Serial.print("ADC fasa S:");
  Serial.print(TeganganFasa_S);
  Serial.print("||");
  Serial.print("ADC fasa T:");
  Serial.print(TeganganFasa_T);
  Serial.println("||");
}
  
```

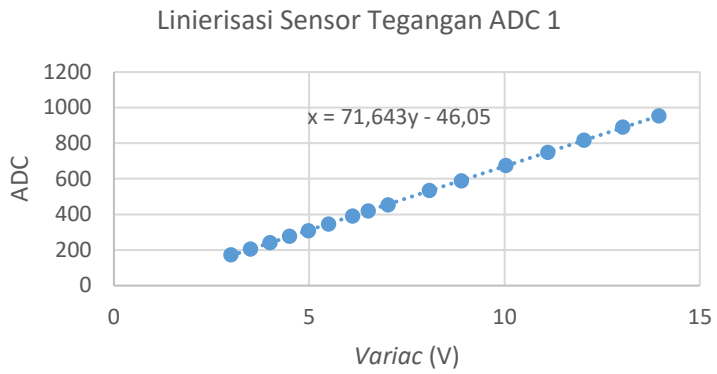
Gambar 4.17 Program Pengujian Sensor Tegangan

Tabel 4.6 Data Linierisasi Sensor Tegangan

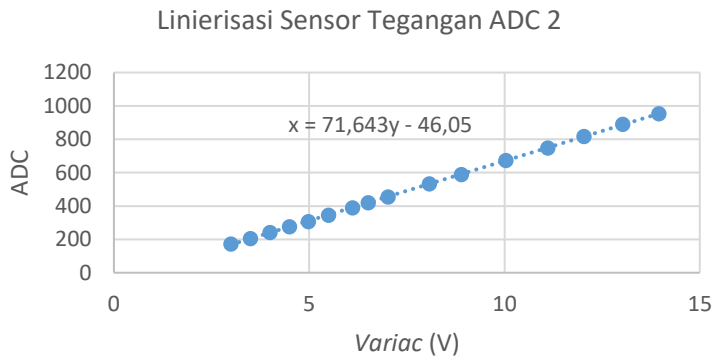
Variac (V)	ADC 1	ADC 2	ADC 3
3	173	173	170
3,5	206	206	203

Variac (V)	ADC 1	ADC 2	ADC 3
4	242	242	238
4,5	277	277	274
4,98	308	308	304
5,5	346	346	341
6,11	390	390	385
6,51	420	420	414
7,02	454	454	447
8,08	534	534	525
8,9	589	589	579
10,04	674	674	663
11,11	749	749	737
12,04	817	817	805
13,03	890	890	875
13,95	954	954	937

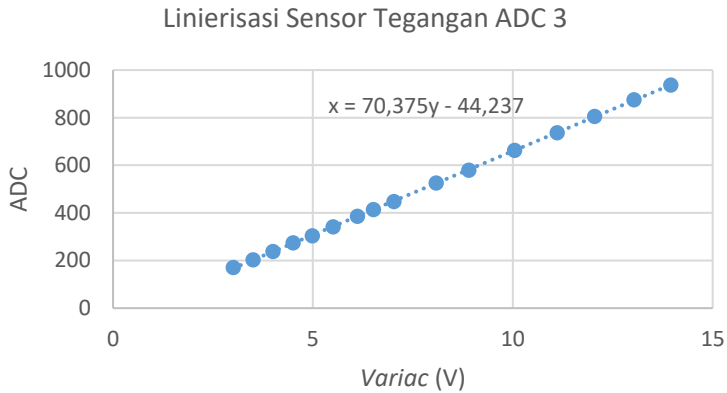
Pada Tabel 4.6 dapat dibuat suatu Persamaan karakteristik linierisasi sensor tegangan pada tiap-tiap fasa yang akan digunakan pembacaan nilai tegangan pada Arduino yang sesuai dengan nilai tegangan pengukuran menggunakan *multimeter* serta menggunakan Persamaan *scatter* pada *Microsoft Excel* yang ditunjukkan linierisasi ADC 1 fasa R pada Gambar 4.18, linierisasi ADC 2 fasa S pada Gambar 4.19, dan linierisasi ADC 3 fasa T pada Gambar 4.20.



Gambar 4.18 Linierisasi Sensor Tegangan ADC 1

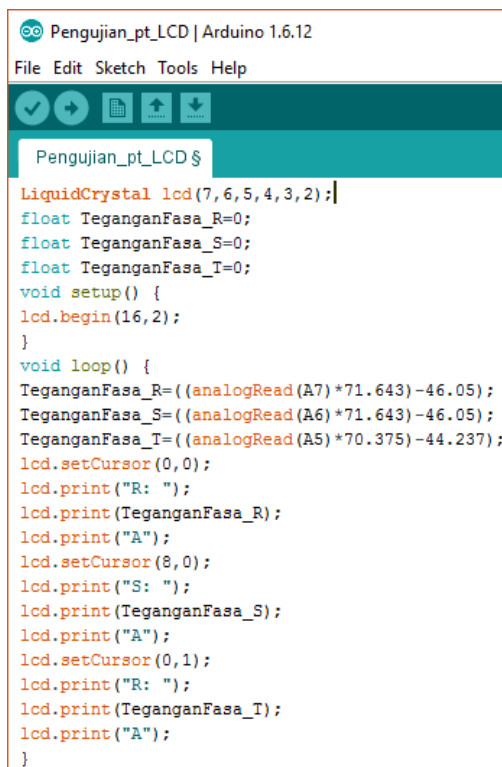


Gambar 4.19 Linierisasi Sensor Tegangan ADC 2



Gambar 4.20 Linierisasi Sensor Tegangan ADC 3

Dari Persamaan linier sensor tegangan yang telah didapatkan pada masing-masing fasa kemudian Persamaan tersebut dimasukkan ke dalam program untuk pengujian pada Gambar 4.21. Hasil pengujian dengan Persamaan tersebut dilihat dengan mengatur *Variac* sehingga tegangan masukan ke Arduino bervariasi. Kemudian pada Tabel 4.7 pada LCD 1 akan menampilkan nilai dari linierisasi fasa R pada Gambar 4.18, LCD 2 akan menampilkan nilai dari linierisasi fasa S pada Gambar 4.19, dan LCD 3 akan menampilkan nilai dari linierisasi fasa T pada Gambar 4.20. Perhitungan presentase *error* pengukuran sensor tegangan didapat dari Persamaan 4.2.



```

Penguian_pt_LCD | Arduino 1.6.12
File Edit Sketch Tools Help

Penguian_pt_LCD $
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
float TeganganFasa_R=0;
float TeganganFasa_S=0;
float TeganganFasa_I=0;
void setup() {
  lcd.begin(16, 2);
}
void loop() {
  TeganganFasa_R=((analogRead(A7)*71.643)-46.05);
  TeganganFasa_S=((analogRead(A6)*71.643)-46.05);
  TeganganFasa_I=((analogRead(A5)*70.375)-44.237);
  lcd.setCursor(0,0);
  lcd.print("R: ");
  lcd.print(TeganganFasa_R);
  lcd.print("A");
  lcd.setCursor(8,0);
  lcd.print("S: ");
  lcd.print(TeganganFasa_S);
  lcd.print("A");
  lcd.setCursor(0,1);
  lcd.print("I: ");
  lcd.print(TeganganFasa_I);
  lcd.print("A");
}

```

Gambar 4.21 Program Tampilan LCD Penguian Sensor Tegangan

Tabel 4.7 Data Penguian Sensor Tegangan

<i>Variac</i> (V)	LCD 1 (V)	<i>Error</i> (%)	LCD 2 (V)	<i>Error</i> (%)	LCD 3 (V)	<i>Error</i> (%)
3,39	3,4	0,2	3,4	0,2	3,38	0,29
4,5	4,51	0,2	4,49	0,2	4,48	0,4
5	5,01	0,2	5,01	0,2	4,97	0,6
6,12	6,13	0,16	6,12	0	6,08	0,6
7,8	7,81	0,12	7,8	0	7,76	0,5
9,48	9,48	0	9,49	0,1	9,46	0,2

<i>Variac</i> (V)	LCD 1 (V)	<i>Error</i> (%)	LCD 2 (V)	<i>Error</i> (%)	LCD 3 (V)	<i>Error</i> (%)
10,03	10	0,29	10	0,2	9,92	1,09
11,15	11,16	0,08	11,14	0,08	11,09	0,5
12,83	12,82	0,07	12,8	0,2	12,74	0,7
13,58	13,57	0,07	13,54	0,2	13,47	0,8

Dari data Tabel 4.7 yang didapat dari pengujian sensor tegangan antara nilai tegangan pada alat ukur dengan nilai tegangan yang tampil pada LCD pada Gambar 4.22, dapat dihitung rata-rata *error* pada sensor tegangan LCD 1 sebesar 0,15%, sensor tegangan LCD 2 sebesar 0,17%, dan sensor tegangan LCD sebesar 0,5%. Perbedaan dari keluaran sensor tersebut dapat dikarenakan faktor internal maupun faktor eksternal. Faktor internal berasal dari karakteristik atau nilai tiap komponen yang tidak semuanya memiliki standar yang baik. Faktor eksternal berasal dari faktor kesalahan pembacaan alat ukur.



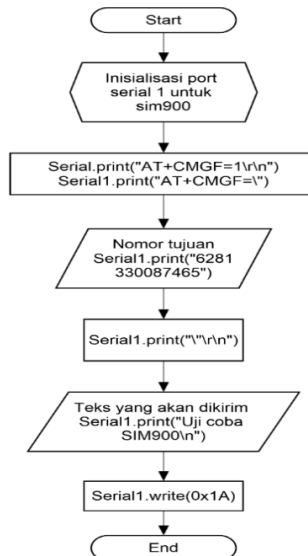
Gambar 4.22 Kalibrasi Sensor Tegangan

4.7 Pengujian Komunikasi

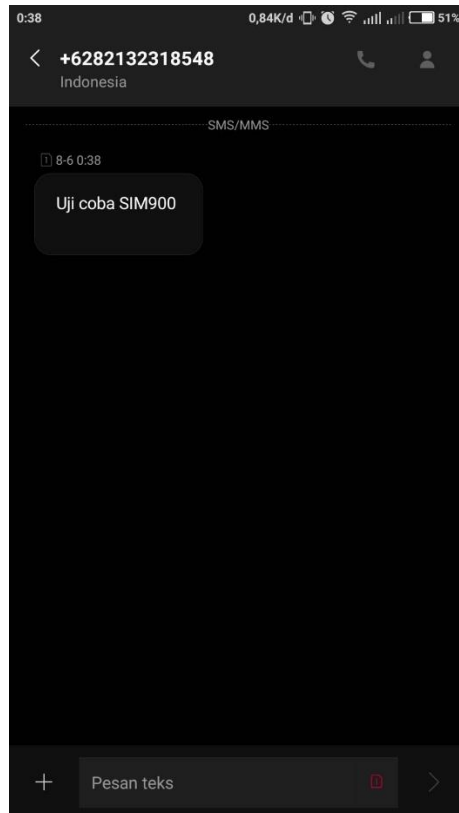
Pada pengujian komunikasi ini terdiri dari pengujian SIM900 dan pengujian Modem *Maestro*. SIM900 pada alat tugas akhir ini berfungsi sebagai pengirim laporan gangguan yang terjadi dan terletak didalam panel alat Tugas Akhir ini. Sedangkan Modem *Maestro* pada alat tugas akhir ini berfungsi sebagai penerima laporan gangguan yang terjadi dan terletak di komputer *server*.

4.7.1 Pengujian SIM900

Dalam melakukan pengujian SIM900 kita perlu menggunakan program *AT+command*. SIM900 yang akan digunakan untuk pengiriman SMS maka SIM900 menggunakan format teks dengan perintah *AT+cmgf=1*, mengatur nilai 1 pada *AT+cmgf*. Perintah *AT+cmgs* merupakan perintah untuk pengiriman SMS pada modul SIM900. *AT+cmgs* diikuti dengan penulisan nomor yang akan dituju. Pengujian SIM900 dengan melakukan pengiriman karakter “Uji coba SIM900” dengan *flowchart* pemrograman di tunjukkan pada Gambar 4.23 dan hasil dari pengiriman yang ditunjukkan pada Gambar 4.24.



Gambar 4.23 *Flowchart* Pengujian SMS SIM900



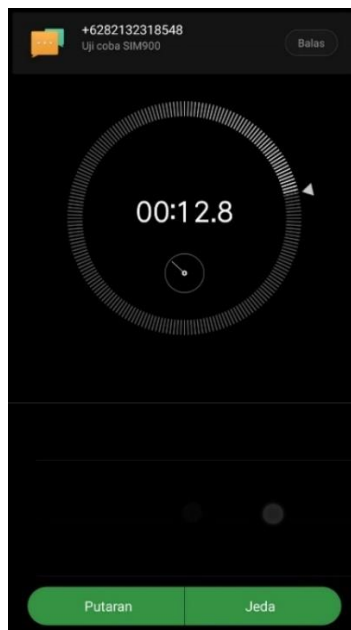
Gambar 4.24 Hasil Pengujian SIM900

Pada tugas akhir ini SMS yang dikirim berupa pemberitahuan ke operator teknis di lapangan bahwa terjadi gangguan pada SUTM penyulang tertentu. Tingkat keberhasilan pengiriman SMS pada tugas akhir ini ditunjukkan dengan selisih waktu pengiriman dengan penerimaan SMS. Hasil pengujian waktu pengiriman dilihat pada Tabel 4.8 dan hasil pengujian waktu pengiriman SMS dilihat pada Gambar 4.25.

Tabel 4.8 Hasil Pengujian SIM900

No	Waktu Pengiriman SMS	Waktu Pengiriman perintah <i>reset</i>
1	00 : 12 : 08 detik	00 : 09 : 08 detik
2	00 : 10 : 17 detik	00 : 07 : 17 detik
3	00 : 09 : 55 detik	00 : 04 : 42 detik
4	00 : 09 : 06 detik	00 : 04 : 10 detik
5	00 : 09 : 15 detik	00 : 05 : 13 detik

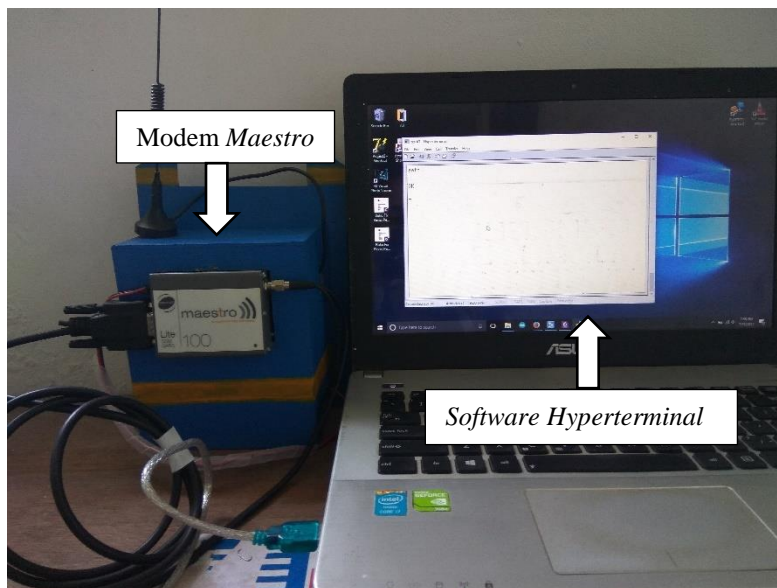
Pada pengujian komunikasi SIM900 waktu pengiriman dan waktu yang di terima Arduino dengan rata-rata waktu pengiriman data gangguan 00:15:36 detik dan rata-rata Arduino penerima SMS yakni selama 00:06:10 detik. Waktu pada pengujian tersebut sudah termasuk waktu pembacaan ADC dan *delay* yang ada pada Arduino. Waktu tersebut juga tergantung pada kekuatan sinyal pada *provider* yang digunakan.



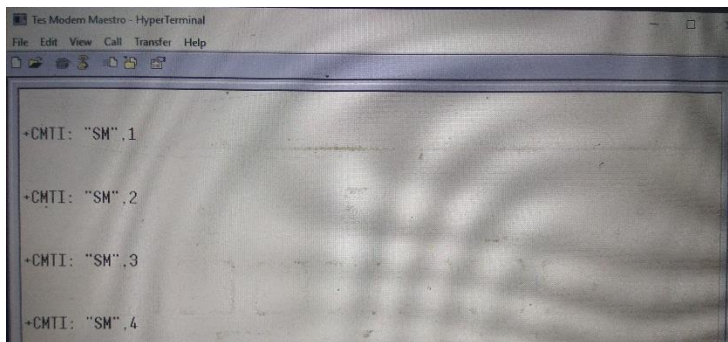
Gambar 4.25 Hasil Pengujian Waktu SMS

4.7.2 Pengujian Modem *Maestro*

Pada pengujian modem *Maestro* Gambar 4.26 ini digunakan *software Hyperterminal* untuk membantu pembacaan sms yang masuk ke modem *Maestro*. Untuk memberikan perintah ke modem *Maestro* digunakan AT+command. Indikasi adanya SMS masuk ke modem *Maestro* adalah dengan munculnya kode “+CMTI: “SM”,1” dapat dilihat pada Gambar 4.27, “+CMTI” menandakan adanya sms masuk, “SM” menandakan SMS disimpan di memori *sim card*, dan “1” menandakan nomor urut SMS tersimpan. Pengujian modem *Maestro* dengan mengirimkan karakter “Uji modem maestro” dari *smartphone* ke nomor tujuan yang terpasang pada modem *Maestro*.



Gambar 4.26 Pengujian Modem *Maestro*



Gambar 4.27 Hasil Pengujian Modem *Maestro*

Pada tugas akhir ini SMS yang diterima berupa pemberitahuan ke operator teknis di lapangan bahwa terjadi gangguan pada SUTM penyulang tertentu. Tingkat keberhasilan penerimaan SMS pada tugas akhir ini ditunjukkan dengan waktu penerimaan dan masuk tidaknya SMS. Hasil pengujian waktu penerimaan dilihat pada Tabel 4.9.

Tabel 4.9 Hasil Pengujian Modem *Maestro*

No	Waktu Penerimaan SMS	SMS Masuk/Tidak
1	00 : 10 : 08 detik	Masuk
2	00 : 09 : 04 detik	Masuk
3	00 : 09 : 41 detik	Masuk
4	00 : 08 : 32 detik	Masuk
5	-	Tidak

Pada pengujian modem *Maestro* didapatkan hasil rata-rata waktu penerimaan data gangguan 00:09:22 detik dan persen *error* SMS yang berhasil masuk sebesar 20% dari 5 kali percobaan. Waktu tersebut juga tergantung pada kekuatan sinyal pada *provider* yang digunakan.

Pada saat presentasi alat tugas akhir yang bertempat di Laboratorium Elektronika Terapan AA104, modem *Maestro* yang digunakan mengalami masalah berupa gagalnya sms yang masuk. Kegagalan tersebut mungkin dikarenakan adanya beberapa faktor yakni:

1. SMS *Pending* akibat buruknya sinyal *provider* Telkomsel yang digunakan.
2. Koneksi kabel modem dengan Laptop yang longgar.
3. Peralatan modem yang kurang baik kondisinya Karena yang digunakan adalah peralatan bekas.

4.8 Pengujian *Short Circuit* dan *Open Circuit Transformer*

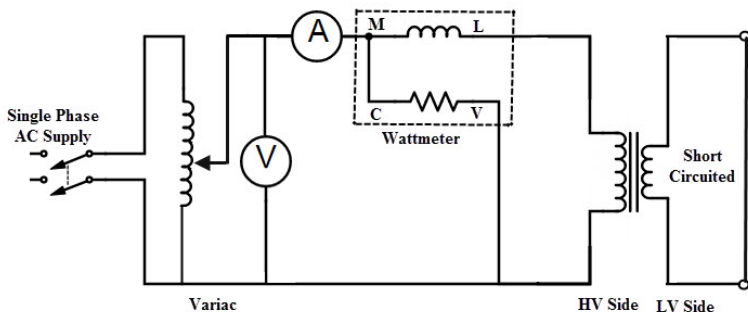
Pengujian *Short Circuit* dan *Open Circuit Transformer* ini bertujuan untuk mengetahui berapa kuat ketahanan trafo dalam mengalirkan daya saat kondisi normal atau ideal dan saat kondisi terjadinya hubung singkat. Kapasitas arus maksimal yang di miliki oleh trafo ini harus diketahui terlebih dahulu sebelum menggunakan trafo. Pada *transformator* yang digunakan yakni *transformator Step Down* 380V/12V dimana maksudnya adalah trafo penurun tegangan dari 380 V primer ke 12 V sekunder. Tujuan lain dari rangkaian pengujian pada trafo ini yakni.

1. Untuk menentukan arus maksimum dan minimum hubung singkat.
2. Untuk menentukan arus gangguan tak simetris bagi gangguan satu dan dua line ke tanah, gangguan line ke line, dan rangkaian terbuka
3. Penyelidikan operasi rele-rele proteksi
4. Untuk menentukan kapasitas pemutus dari *circuit breaker*
5. Untuk menentukan distribusi arus gangguan dan tingkat tegangan busbar selama gangguan

Berikut adalah langkah-langkah dalam melakukan pengujian *Short Circuit* trafo.

1. Tegangan *V input* awal adalah 0 Volt
2. Terminal disisi sekunder disambungkan (*Short Circuit*)
3. Naikan tegangan *V Input* dengan mengatur potensiometer, sampai arus yang terbaca pada Ampermeter disisi sekunder mencapai *I rate*.

Hanya beberapa data saja yang akan diambil pada proses pengujian trafo ini diantaranya yakni bisa dilihat pada Tabel 4.10. Skema pengujian *Short Circuit* trafo dapat dilihat pada Gambar 4.28.



Gambar 4.28 Skema Pengujian *Short Circuit* Trafo

Tabel 4.10 Hasil Pengujian Trafo

Fasa	<i>Open Circuit Test</i>			<i>Short Circuit Test</i>		
	Voc (V)	Ioc (A)	Poc (W)	Vsc (V)	Isc (A)	Psc (W)
R	380	0,05	13	42	1,5	60
S	380	0,04	8,5	41	1,5	55
T	380	0,055	11,5	40	1,5	55

4.8.1 Perhitungan Impedansi Dalam Trafo

Dari pengujian *transformator* yang telah dilakukan dapat diambil data berupa Tegangan *Open Circuit* , Arus *Open Circuit* , dan Daya *Open Circuit* dari *Open Circuit Test* , sedangkan pada *Short Circuit Test* diambil data berupa Tegangan *Short Circuit* , Arus *Short Circuit* , dan Daya *Short Circuit* . Setelah itu data yang telah diambil akan dimasukkan ke dalam Persamaan berikutnya untuk mengambil data lain yang diperlukan. Langkah berikutnya yakni mencari impedansi dalam dan juga impedansi ekuivalen yang dimiliki oleh trafo. Tujuan mencari nilai Impedansi dalam pada trafo ini adalah untuk menentukan perhitungan arus hubung singkat yang diakibatkan oleh trafo bila sudah tersambung dengan jaringan nanti. Beberapa data yang dicari yakni Tahanan *Circuit* (R_c) pada Persamaan 4.3, Daya Semu (S) pada Persamaan 4.4, Daya Reaktif (Q_{oc}) pada Persamaan 4.5, dan juga Induktansi Dalam (X_m) pada Persamaan 4.6. Kemudian hasil dari perhitungan dapat dilihat di Tabel 4.11.

$$R_c = \frac{V_{oc}^2}{P_{oc}} \dots\dots\dots (4.3)$$

$$S = V_{oc} \times I_{oc} \dots\dots\dots (4.4)$$

$$Q_{oc} = \sqrt{S^2 - P_{oc}^2} \dots\dots\dots (4.5)$$

$$X_m = \frac{V_{oc}^2}{Q_{oc}} \dots\dots\dots (4.6)$$

Tabel 4.11 Data Impedansi Dalam Trafo

Impedansi Dalam			
Rc (Ohm)	S (Watt)	Qoc (Ohm)	Xm (Ohm)
11107,69	19	13,85	10421,17
16988,23	15,2	12,6	11459,23
12556,52	20,9	17,45	8274,29

4.8.2 Perhitungan Impedansi Ekivalen Trafo

Selain data Impedansi Dalam trafo kita juga harus mencari data Impedansi Ekivalen dari trafo. Beberapa data yang dicari yakni Resistansi Ekivalen (R_{eq}), Impedansi Ekivalen (Z_{eq}), lalu Induktansi Ekivalen (X_{eq}). Data tersebut dapat dicari dari Persamaan 4.7, Persamaan 4.8, dan Persamaan 4.9. Kemudian hasil dari perhitungan dapat dilihat di Tabel 4.12.

$$R_{eq} = \frac{P_{sc}}{I_{sc}^2} \dots\dots\dots (4.7)$$

$$Z_{eq} = \frac{V_{sc}}{I_{sc}} \dots\dots\dots (4.8)$$

$$X_{eq} = \sqrt{Z_{eq}^2 - R_{eq}^2} \dots\dots\dots (4.9)$$

Tabel 4.12 Data Impedansi Ekivalen Trafo

Impedansi Ekivalen		
R_{eq} (Ohm)	Z_{eq} (Ohm)	X_{eq} (Ohm)
26,6	28	8,5
24,4	27,3	12,2
24,4	26,6	10,6

4.8.3 Perhitungan Per Unit (PU)

Setelah didapat data Impedansi Ekivalen dan juga Impedansi Dalam trafo maka kedua data tersebut akan dipakai dalm perhitungan data Per Unit (PU). Data PU ini digunakan untuk membuat satuan menjadi sama dari data-data yang telah diambil. Dengan data PU ini kita dapat mencari Daya Semu (S), Tahanan *Circuit* (Rc), Daya Reaktif (Qoc), dan Resistansi Ekivalen (Req), Impedansi Ekivalen (Zeq), lalu Induktansi Ekivalen (Xeq) dari trafo yang telah diuji. Dengan menggunakan Persamaan 4.10 pertama kita harus mencari nilai *Zbase* terlebih dahulu. Kemudian hasil perhitungan PU bisa dilihat di Tabel 4.13.

$$Z_{base} = \frac{V^2}{MVA} \dots\dots\dots(4.10)$$
$$Z_{base} = \frac{380^2}{1000}$$
$$Z_{base} = 144,4 \text{ ohm}$$

Dari Persamaan 4.10 setelah nilai *Zbase* didapat maka dapat dimasukan ke dalam Persamaan 4.11 untuk mencari nilai PU.

$$PU = \frac{Req}{Z_{base}} = \frac{Zeq}{Z_{base}} = \frac{Xeq}{Z_{base}} = \frac{Xm}{Z_{base}} = \frac{Rc}{Z_{base}} \dots\dots\dots (4.11)$$

Tabel 4.13 Data Per Unit Trafo

Fasa	Per Unit					
	Rc (Ohm)	S (Watt)	Qoc (Ohm)	Req (Ohm)	Zeq (Ohm)	Xeq (Ohm)
R	76,9	0,019	0,014	72,2	0,193	0,064
S	117,6	0,015	0,013	79,4	0,189	0,084
T	86,9	0,021	0,017	57,3	0,184	0,073

4.9 Pengujian Arus Hubung Singkat

Pada pengujian arus hubung singkat atau *Short Circuit* disini kita akan menguji modul gangguan yang akan digunakan untuk merepresentasikan lokasi gangguan pada Gambar 4.29. Pengujian ini dilakukan agar kita dapat mengetahui seberapa nilai arus hubung singkat satu fasa ke tanah dan juga dua fasa yang ada pada modul gangguan. Tujuan dari pengujian ini adalah untuk mengetahui berapa nilai MCB

Diagram illustrating a three-phase power distribution system. The system includes a 380/12 V transformer, two MCBs (Main Circuit Breakers), and a busbar system. The phases are labeled 3Ø. The system is grounded (M). Two 15 W loads are connected to the yellow phase.

		Easo	
--	--	-------------	--

No	Jarak	Rasa			Rata-rata (A)
		R (A)	S (A)	T (A)	
1	1 KM	11,7	12,1	11,8	11,86
2	2 KM	7,6	7,9	7,6	7,7
3	3 KM	5,3	5,5	5,4	5.4
4	4 KM	4,97	5,3	4,94	5.07

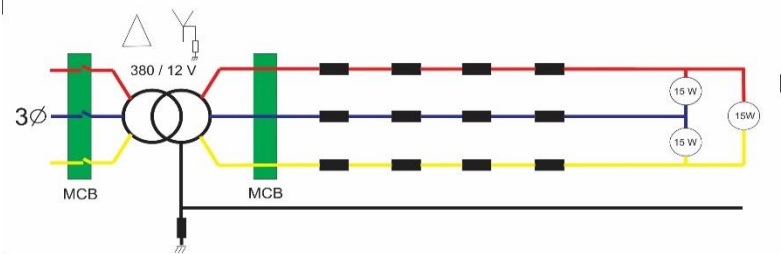
		P	S	S
--	--	----------	----------	----------

No	Jarak	R-S		S-T		Rata-rata (A)
		R (A)	S (A)	S (A)	T (A)	
1	1 KM	10,14	10,4	10,35	10,5	10,34
2	2 KM	8,6	8,42	8,6	8,73	8,58
3	3 KM	6,8	6,31	6,5	6,71	6,58
4	4 KM	4,98	5,1	4,89	4,97	4,98

9 10 11 12

4.10 Pengujian Modul Gangguan

Pengujian modul gangguan ini dilakukan untuk mencari nilai arus yang mengalir dan juga tegagan yang ada pada modul gangguan dan juga beban yang dipakai. Pengujian ini diperlukan agar kita dapat menganalisa kerja modul gangguan agar sesuai dengan yang kita rancang dan sudah siap dilakukan uji keseluruhan sistem. Modul gangguan ini menggunakan *resistor* 0,33 *Ohm* berjumlah 12 buah yang diseri 4 buah per fasa untuk mewakili jark 4 KM. Kemudian beban yang digunakan yakni 3 buah lampu halogen 15 *Watt*, untuk MCB yang digunakan adalah jenis MCB 3 fasa dengan nominal arus 6 A. Skema pengujian rangkaian dapat dilihat pada Gambar 4.30.



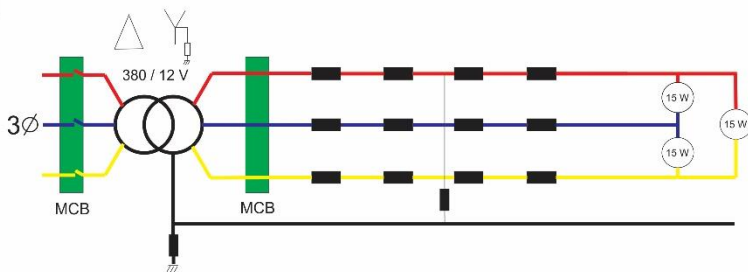
Gambar 4.30 Skema Pengujian Rangkaian

Tabel 4.16 Data Pengujian Saluran Modul Gangguan

No	Fasa	Saluran (A)	V p-n (V)	V p-p (V)	V Saluran (V)
1	R	2,33	7,22	-	2,8
2	S	2,30	6,95	-	3,1
3	T	2,35	7,3	-	3,1
4	R – S	-	-	12,26	-
5	R – T	-	-	12,5	-
6	S – T	-	-	12,2	-

Tabel 4.17 Data Pengujian Beban Modul Gangguan

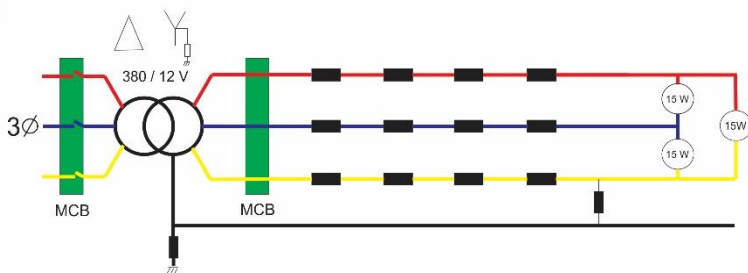
No	Beban	V beban (V)
1	1	4,06
2	2	4,52
3	3	4,32



Gambar 4.32 Hubung Singkat 2 KM

Tabel 4.20 Data Pengujian Hubung Singkat 2 KM

I sc (A)	V sc (V)	V beban (V)		
		1	2	3
7,2	6,87	2,18	4,38	2,4



Gambar 4.33 Hubung Singkat 4 KM

Tabel 4.21 Data Pengujian Hubung Singkat 4 KM

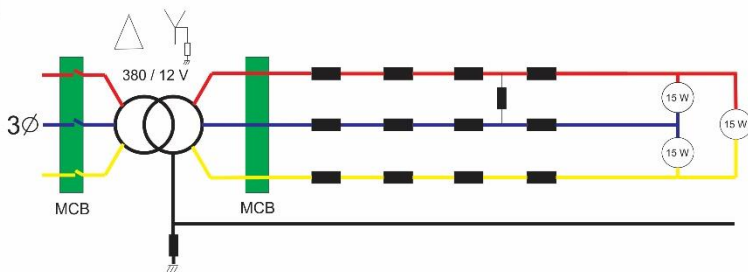
I sc (A)	V sc (V)	V beban (V)		
		1	2	3
5,8	7,1	3,7	2,5	2,1

Pada pengujian gangguan satu fasa ke tanah dari Tabel 4.19 Data Pengujian Hubung Singkat 1 KM dapat diketahui bahwa arus hubung singkat yang terjadi sebesar 11,6 A dan tegangan *drop* menjadi 6,3 V, jika

4.10.2 Pengujian Modul Gangguan Antar Fasa

Tabel 4.22 Data Pengujian Hubung Singkat Fasa S dan T

98



Gambar 4.35 Hubung Singkat Fasa R dan S

Tabel 4.23 Data Pengujian Hubung Singkat Fasa R dan S

I _{sc} (A)		V _{sc} (V)		V Beban (V)		
S	R	S	R	1	2	3
4,9	4,85	6,9	7,15	0	3,7	3,6

Pada pengujian gangguan antar fasa dari Tabel 4.22 Data Pengujian Hubung Singkat Fasa S dan T dapat diketahui bahwa arus hubung singkat yang terjadi di fasa S sebesar 5,9 A dan di fasa T sebesar 5,6 A dan tegangan *drop* di fasa S sebesar 6,9 V dan di fasa T sebesar 7,2 V, jika dibandingkan dengan arus hubung singkat dan tegangan *drop* yang diketahui dari Tabel 4.23 Data Pengujian Hubung Singkat fasa R dan S, arus hubung singkat yang terjadi semakin kecil dikarenakan jarak gangguan yang semakin jauh. Sedangkan tegangan *drop* yang terjadi semakin besar. Dari data yang diperoleh maka dapat diambil kesimpulan saat terjadi gangguan besar arus dan tegangan akan berbanding terbalik.

4.11 Pengujian Keseluruhan

Ketika dilakukan pengujian secara keseluruhan, dilakukan seluruh koordinasi semua sensor, prototipe jaringan dan tampilan pada aplikasi Android. Sistem menggunakan jaringan tiga fasa dengan indikator berupa tiga buah lampu sebesar 15 *Watt* pada tiap fasa R, S, dan T. Pengujian sistem secara keseluruhan dilaksanakan pada :

Waktu : Kamis, 13 Juli 2017, 01.30 WIB

Tempat : Laboratorium Tegangan Tinggi, Teknik Elektro ITS

Pada modul gangguan beban menyala terang semua karena tidak ada gangguan yang terjadi, sedangkan pada Sistem tidak menampilkan besar arus dalam keadaan normal (tidak ada arus gangguan) sehingga hanya berupa tampilan tanggal dan waktu *real time* pada LCD dan status normal tiap fasa pada LCD pada Gambar 4.36 dan Gambar 4.37.



Gambar 4.36 Modul Gangguan pada Kondisi Normal



Gambar 4.37 Tampilan LCD saat Keadaan Normal

4.11.1 Gangguan Fasa S

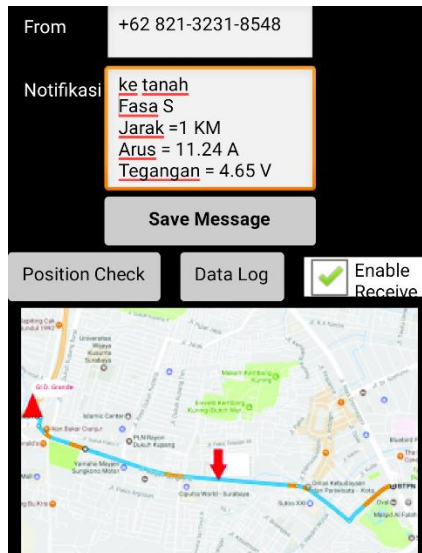
Pada pengujian gangguan fasa S, pengujian kondisi ini dilakukan dengan menekan sakelar 1 untuk menimbulkan gangguan, hasil dari proses pengujian pada sakelar 1 dapat dilihat pada Gambar 4.38 sampai Gambar 4.41.



Gambar 4.38 Modul Gangguan saat Gangguan Fasa S



Gambar 4.39 Panel saat Gangguan Fasa S



Gambar 4.40 Gangguan Fasa S pada Aplikasi Android

Pelaporan Gangguan
Jaringan Tegangan Menengah 20 KV

Fasa Terganggu

R S T

Records

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangguan

Rincian Gangguan

Tanggal: 15/7/1
Waktu: 4.00.0
Jenis Gangguan: Gangguan satu fasa ke tanah
Fasa Yang Terganggu: Fasa S
Jarak Gangguan: 1 KM

Connection: Set Port Close

Clear Save

Gambar 4.41 Gangguan Fasa S pada Aplikasi *Server*

Pada saat gangguan fasa S dapat dilihat pada modul gangguan tegangan pada beban mengalami drop tegangan sehingga menandakan bahwa sedang terjadi gangguan hubung singkat pada Gambar 4.38. Tampilan pada layar LCD pada Gambar 4.39 ketika gangguan terjadi akan menampilkan nilai arus sebesar 11,24 A dan jarak 1 km dan indikator fasa S mengalami gangguan begitu juga pada tampilan *text box* aplikasi Android pada Gambar 4.40 dan juga di aplikasi *server* pada Gambar 4.41.

Pada tampilan aplikasi Android ditampilkan fasa S yang mengalami gangguan, nilai arus sebesar 11,24 A, tegangan 4,65 V dan jarak 1 km. Pengujian untuk fasa S pada sakelar 1 dilakukan sebanyak 7 kali percobaan. Untuk hasil pengujian yang lebih lengkap dapat dilihat pada Tabel 4.24

Tabel 4.24 Hasil Pengujian Gangguan Fasa S

Fasa S				
No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
1	11,58	4,75	1	00:04.5

No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
2	11,45	4,75	1	00:05.2
3	11,16	4,82	1	00:04.9
4	10,23	4,66	1	00:06.3
5	10,14	4,5	1	00:05.5
6	10,1	4,55	1	00:10.5
7	10,2	4,53	1	00:06.3
Rata-rata	10,69	4,65	1	00:06.1

4.11.2 Gangguan Fasa R

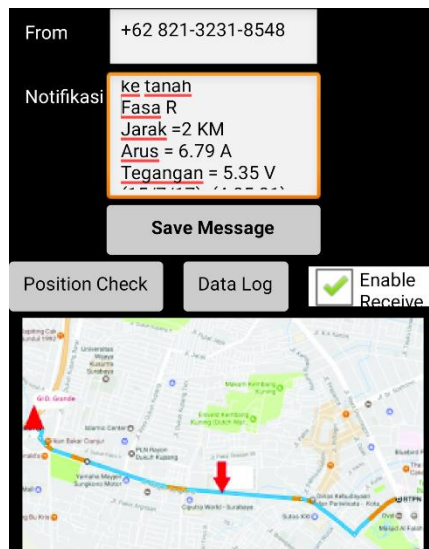
Pada pengujian gangguan fasa R, pengujian kondisi ini dilakukan dengan menekan sakelar 2 untuk menimbulkan gangguan, hasil dari proses pengujian pada sakelar 2 dapat dilihat pada Gambar 4.42 sampai Gambar 4.45.



Gambar 4.42 Modul Gangguan saat Gangguan Fasa R



Gambar 4.43 Panel Gangguan Fasa R



Gambar 4.44 Gangguan Fasa R pada Aplikasi Android

Pelaporan Gangguan
Jaringan Tegangan Menengah 20 KV

Fasa Terganggu

R S T

Records

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangguan
15/7/1	4:00:0	Gangguan satu fasa ke tan	Fasa S	1 KM

Rincian Gangguan

Tanggal: 15/7/1
Waktu: 4:07:5
Jenis Gangguan: Gangguan satu fasa ke tanah
Fasa Yang Terganggu: Fasa R
Jarak Gangguan: 2 KM

Connection: Set Port Close

Clear Save

Gambar 4.45 Gangguan Fasa R pada Aplikasi *Server*

Pada saat gangguan fasa R dapat dilihat pada modul gangguan tegangan pada beban mengalami drop tegangan sehingga menandakan bahwa sedang terjadi gangguan hubung singkat pada Gambar 4.42. Tampilan pada layar LCD pada Gambar 4.43 ketika gangguan terjadi akan menampilkan nilai arus sebesar 6,79 A dan jarak 2 km dan indikator fasa R mengalami gangguan begitu juga pada tampilan *text box* aplikasi Android pada Gambar 4.44 dan juga di aplikasi *server* pada Gambar 4.45.

Pada tampilan aplikasi Android ditampilkan fasa S yang mengalami gangguan, nilai arus sebesar 6,79 A, tegangan 5,35 V dan jarak 2 km. Pengujian untuk fasa R pada sakelar 2 dilakukan sebanyak 7 kali percobaan. Untuk hasil pengujian yang lebih lengkap dapat dilihat pada Tabel 4.25

Tabel 4.25 Hasil Pengujian Gangguan Fasa R

Fasa R				
No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
1	6,59	5,34	2	00:05.2

No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
2	6,79	5,35	2	00:06.3
3	6,79	5,41	2	00:04.7
4	6,74	5,4	2	00:08.3
5	6,68	5,33	2	00:10.9
6	6,83	5,44	2	00:15.5
7	6,78	5,36	2	00:16.1
Rata-rata	6,74	5,38	2	00:09.5

4.11.3 Gangguan Fasa T

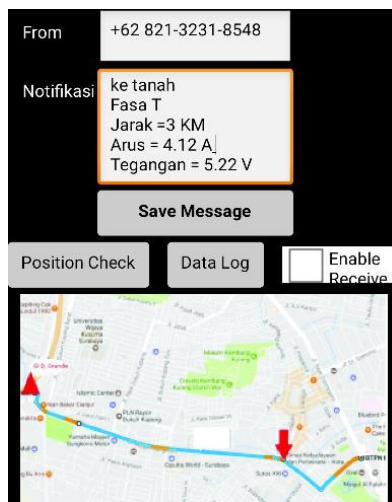
Pada pengujian gangguan fasa T, pengujian kondisi ini dilakukan dengan menekan sakelar 4 untuk menimbulkan gangguan, hasil dari proses pengujian pada sakelar 4 dapat dilihat pada Gambar 4.46 sampai Gambar 4.49.



Gambar 4.46 Modul Gangguan saat Gangguan Fasa T



Gambar 4.47 Panel Gangguan Fasa T



Gambar 4.48 Gangguan Fasa T pada Aplikasi Android

Pelaporan Gangguan
Jaringan Tegangan Menengah 20 KV

Fasa Terganggu

Rincian Gangguan

Tanggal: 15/7/1
Waktu: 4:13:1
Jenis Gangguan: Gangguan satu fasa ke tanah
Fasa Yang Terganggu: Fasa T
Jarak Gangguan: 3 KM

Records

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangguan
15/7/1	4:00:0	Gangguan satu fasa ke tanah	Fasa S	1 KM
15/7/1	4:07:5	Gangguan satu fasa ke tanah	Fasa R	2 KM

Connection: Set Port, Close, Clear, Save

Gambar 4.49 Gangguan Fasa T pada Aplikasi *Server*

Pada saat gangguan fasa T dapat dilihat pada modul gangguan tegangan pada beban mengalami drop tegangan sehingga menandakan bahwa sedang terjadi gangguan hubung singkat pada Gambar 4.46. Tampilan pada layar LCD pada Gambar 4.47 ketika gangguan terjadi akan menampilkan nilai arus sebesar 4,12 A dan jarak 3 km dan indikator fasa T mengalami gangguan begitu juga pada tampilan *text box* aplikasi Android pada Gambar 4.48 dan juga di aplikasi *server* pada Gambar 4.49.

Pada tampilan aplikasi Android ditampilkan fasa S yang mengalami gangguan, nilai arus sebesar 4,12 A, tegangan 5,22 V dan jarak 3 km. Pengujian untuk fasa T pada sakelar 4 dilakukan sebanyak 7 kali percobaan. Untuk hasil pengujian yang lebih lengkap dapat dilihat pada Tabel 4.26

Tabel 4.26 Hasil Pengujian Gangguan Fasa T

Fasa T				
No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
1	4,12	5,22	3	00:05.3
2	4,06	5,15	3	00:06.8

No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
3	4,05	5,15	4	00:04.6
4	4,1	5,2	4	00.05.6
5	4,2	5,2	4	00:09.4
6	4,11	5,17	4	00:08.5
7	4,04	5,1	3	00:10.1
Rata-rata	4,10	5,17	3,6	00:07.1

4.11.4 Gangguan Fasa S – T

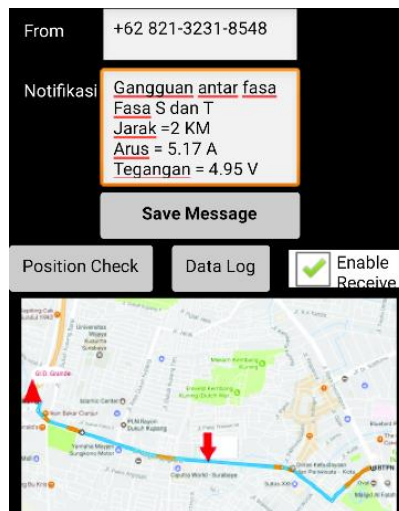
Pada pengujian gangguan fasa S - T, pengujian kondisi ini dilakukan dengan menekan sakelar 3 untuk menimbulkan gangguan, hasil dari proses pengujian pada sakelar 3 dapat dilihat pada Gambar 4.50 sampai Gambar 4.53.



Gambar 4.50 Modul Gangguan saat Gangguan Fasa S-T




Gambar 4.51 Panel saat Gangguan S – T




Gambar 4.52 Gangguan Fasa S-T pada Aplikasi Android

Form1



Pelaporan Gangguan

Jaringan Tegangan Menengah 20 KV



Fasa Terganggu

R

S

T

Records

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangguan
15/7/1	4:00:0	Gangguan satu fasa ke tan	Fasa S	1 KM
15/7/1	4:07:5	Gangguan satu fasa ke tan	Fasa R	2 KM
15/7/1	4:13:1	Gangguan satu fasa ke tan	Fasa T	3 KM

Rincian Gangguan

Tanggal

Waktu

Jenis Gangguan

Fasa Yang Terganggu

Jarak Gangguan

Connection

Gambar 4.53 Gangguan S-T pada Aplikasi *Server*

Pada saat gangguan fasa S - T dapat dilihat pada modul gangguan tegangan pada beban mengalami drop tegangan sehingga menandakan bahwa sedang terjadi gangguan hubung singkat pada Gambar 4.50. Tampilan pada layar LCD pada Gambar 4.51, ketika gangguan terjadi akan menampilkan nilai arus sebesar 5,17 A dan jarak 2 km dan indikator fasa S - T mengalami gangguan begitu juga pada tampilan *text box* aplikasi Android pada Gambar 4.52 dan juga di aplikasi *server* pada Gambar 4.53.

Pada tampilan aplikasi Android ditampilkan fasa S - T yang mengalami gangguan, nilai arus sebesar 5,17 A, tegangan 4,95 V dan jarak 2 km. Pengujian untuk fasa S - T pada sakelar 3 dilakukan sebanyak 7 kali percobaan. Untuk hasil pengujian yang lebih lengkap dapat dilihat pada Tabel 4.27.

Tabel 4.27 Hasil Pengujian Gangguan Fasa S - T

Fasa S - T				
No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
1	5,17	4,95	2	00:06.2

No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
2	5,25	4,98	2	00:06.7
3	5,3	5,05	2	00:06.4
4	5,15	4,91	2	00:08.6
5	5,23	4,96	2	00:05.7
6	5,19	5	2	00:06.9
7	5,15	4,88	2	00:07.9
Rata-rata	5,21	4,96	2	00:06.9

4.11.5 Gangguan Fasa R - S

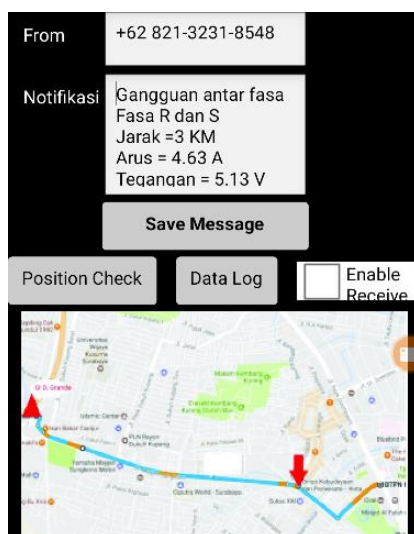
Pada pengujian gangguan fasa R - S, pengujian kondisi ini dilakukan dengan menekan sakelar 5 untuk menimbulkan gangguan, hasil dari proses pengujian pada sakelar 5 dapat dilihat pada Gambar 4.54 sampai Gambar 4.57.



Gambar 4.54 Modul Gangguan saat Gangguan Fasa R-S



Gambar 4.55 Panel Gangguan saat Gangguan R – S



Gambar 4.56 Gangguan R-S pada Aplikasi Android

**Pelaporan Gangguan
Jaringan Tegangan Menengah 20 KV**

Fasa Terganggu

R S T

Records

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangguan
15/7/1	4:18.2	Gangguan antar fasa	Fasa S dan T	2 KM
15/7/1	4:21.0	Gangguan satu fasa ke tanah	Fasa S	1 KM
15/7/1	4:22.0	Gangguan satu fasa ke tanah	Fasa R	2 KM

Rincian Gangguan

Tanggal: 15/7/1
Waktu: 4:31.1
Jenis Gangguan: Gangguan antar fasa
Fasa Yang Terganggu: Fasa R dan S
Jarak Gangguan: 3 KM

Connection

Gambar 4.57 Gangguan R-S pada Aplikasi *Server*

Pada saat gangguan fasa R - S dapat dilihat pada modul gangguan tegangan pada beban mengalami drop tegangan sehingga menandakan bahwa sedang terjadi gangguan hubung singkat pada Gambar 4.54. Tampilan pada layar LCD pada Gambar 4.55, ketika gangguan terjadi akan menampilkan nilai arus sebesar 4,63 A dan jarak 3 km dan indikator fasa R - S mengalami gangguan begitu juga pada tampilan *text box* aplikasi Android pada Gambar 4.56 dan juga di aplikasi *server* pada Gambar 4.57.

Pada tampilan aplikasi Android ditampilkan fasa R - S yang mengalami gangguan, nilai arus sebesar 4,63 A, tegangan 5,13 V dan jarak 3 km. Pengujian untuk fasa S - R pada sakelar 5 dilakukan sebanyak 7 kali percobaan. Untuk hasil pengujian yang lebih lengkap dapat dilihat pada Tabel 4.28.

Tabel 4.28 Hasil Pengujian Gangguan Fasa R - S

Fasa R-S				
No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
1	4,52	5,06	3	00:04.5

No	Arus (A)	Tegangan (V)	Jarak (KM)	Waktu (S)
2	4,5	5,08	3	00:04.2
3	4,61	5,06	3	00:06.3
4	4,6	5,12	3	00:10.4
5	4,71	5,15	4	00:08.4
6	4,64	5,04	3	00:09.7
7	4,42	5,03	3	00:06.3
Rata-rata	4,57	5,08	3,1	00:07.1

4.12 Database Aplikasi Android dan Delphi

Pada aplikasi ini kami menyediakan media penyimpanan berupa *database*, sehingga pesan laporan gangguan yang diterima pada Android dapat disimpan pada *micro sd* sehingga pesan SMS tadi tidak hilang ketika aplikasi dimatikan. Proses penyimpanan pesan dapat dilakukan dengan menekan tombol *Save message* yang terdapat pada *screen 2*. Sedangkan pada komputer *server* juga diberikan penyimpanan berupa *database* sehingga pelaporan dapat direkapitulasi. Format penyimpanan *file* menggunakan format *file.txt* sebagai media penyimpanan pada *smartphone* yang ditunjukkan pada Gambar 4.58 dan media penyimpanan pada komputer *server* pada Gambar 4.59.

```

Gangguan satu fasa ke tanah
Fasa R
Jarak =2 KM
Arus = 6.73 A
Tegangan = 5.38 V
(16/8/17) (0:01:21)

Gangguan satu fasa ke tanah
Fasa R
Jarak =2 KM
Arus = 6.05 A
Tegangan = 5.52 V
(16/8/17) (0:08:37)

Gangguan antar fasa
Fasa S dan T
Jarak =3 KM
Arus = 4.99 A
Tegangan = 5.08 V
(16/8/17) (0:21:26)

Gangguan antar fasa
Fasa S dan T
Jarak =2 KM
Arus = 5.15 A
Tegangan = 4.84 V
(16/8/17) (0:21:26)

```

Gambar 4.58 Database SMS pada Aplikasi Android

Tanggal	Waktu	Jenis Gangguan	Fasa Yang Terganggu	Jarak Gangguan
16/8/17	1:10:44	Gangguan satu fasa k	Fasa R	2 KM
16/8/17	1:37:43	Gangguan antar fasa	Fasa R dan S	3 KM
16/8/17	2:27:32	Gangguan antar fasa	Fasa R dan S	3 KM
16/8/17	2:29:02	Gangguan satu fasa ke tanah	Fasa T	4 KM
16/8/17	2:31:04	Gangguan satu fasa ke tanah	Fasa S	1 KM
16/8/17	2:32:47	Gangguan antar fasa	Fasa S dan T	3 KM
16/8/17	2:35:49	Gangguan satu fasa ke tanah	Fasa T	4 KM
16/8/17	2:44:58	Gangguan satu fasa ke tanah	Fasa S	1 KM

Gambar 4.59 Database SMS pada Aplikasi Delphi

Berdasarkan pada pengujian alat keseluruhan tersebut telah didapatkan hasil yang cukup baik. Sensor arus dapat membaca arus hubung singkat dan sensor tegangan dapat membaca *drop* tegangan yang terjadi sesaat ketika ada gangguan sebelum *trip*-nya MCB. Data gangguan dapat dikirimkan melalui SMS dan bisa diterima pada Android dan Komputer *Server*. Dari 7 kali pengujian pada setiap gangguan didapatkan hasil rata-rata gangguan pada sakelar fasa 1 km, fasa R 2 km dan fasa S-R 2 km. Sedangkan ketidak akuratan terjadi pada gangguan di fasa T 3,6 km dan fasa S-T 3,1 km. Ketidak akuratan tersebut terjadi dikarenakan oleh arus gangguan yang tergantung dari pembacaan arus tertinggi sebelum MCB mengalami *trip*. Selain itu faktor panasnya *resistor* yang terlalu lama berada dalam kondisi gangguan akan menyebabkan tegangan dan arus yang mengalir akan berubah bila semakin lama panas.

4.13 Analisa Relevansi

Prototype ini merupakan alat peraga dari peralatan yang ada pada jaringan distribusi listrik. Untuk bisa diimplementasikan ke jaringan listrik yang sebenarnya memerlukan analisa implementasi terkait pengembangan komponen pengganti yang sesuai terutama untuk pengembangan sistem pelaporan lokasi gangguan berdasarkan gangguan satu fasa ke tanah pada saluran udara tegangan menengah.

Hasil yang didapat dari alat ini cukup untuk dapat merepresentasikan saat terjadi di kondisi real meskipun bila kita

melakukan proses gangguan hubung singkat akan menyebabkan panas berlebih pada *resistor* yang menyebabkan arus dan tegangan akan semakin turun atau berbeda dari kondisi awal.

Pada pengembangannya di lapangan alat ini dapat menggunakan sensor arus CT yang mempunyai rating lebih besar lagi yang sesuai dengan kondisi arus pada Jaringan Tegangan Menengah 20 kV. Untuk sensor tegangan dapat menggunakan trafo PT *Step Down* yang sesuai dengan kondisi nyata di lapangan.

Jika alat ini digunakan pada sumber penyulang khusus, maka dapat dimanfaatkan sebagai pendeteksi lokasi gangguan pada sistem distribusi jaringan tegangan menengah. Dalam pembuatan ini harus diperhatikan bahwa alat pemutus yang digunakan merupakan MCB *thermal* sehingga alat ini memiliki kekurangan dalam akurasi menentukan lokasi gangguan. Dalam penerapan sistem jaringan distribusi yang sesungguhnya dimana PMT yang digunakan menggunakan prinsip waktu sehingga akurasi yang didapatkan dapat lebih baik.

Diharapkan dengan adanya alat ini dapat membantu PLN dalam memperbaiki kinerja nya khususnya pada faktor perbaikan SAIDI dan SAIFI yang saat ini bila terjadi gangguan, target perbaikan yang ditentukan oleh PLN dalam durasi waktu 3 jam bisa ditingkatkan lagi supaya lebih singkat terutama dalam kegiatan pencarian lokasi gangguan sehingga PLN dapat mengurangi kerugian finansial yang terjadi.

BAB V

PENUTUP

Dari pembuatan ”Prototipe dan Rancang Bangun Sistem Gangguan Impedansi Tinggi Pada SUTM 20 kV dengan Sistem Pelaporan Gangguan Menggunakan Komunikasi Data Modul GSM” dan pengujian dari perangkat keras (*hardware*) hingga perangkat lunak (*software*) dapat ditarik kesimpulan dan saran sebagai berikut.

5.1 Kesimpulan

Dari Pembuatan dan pengujian alat ini mulai dari awal hingga akhir dapat diambil kesimpulan yakni:

1. Alat ini dapat merepresentasikan keadaan pada waktu gangguan yang sesungguhnya. Selain itu sistem jaringan alat ini juga dibuat hampir mirip dengan sistem jaringan distribusi 20 kV PLN dalam keadaan yang nyata.
2. Arus hubung singkat yang terjadi di modul dapat mewakili kondisi saat arus hubung singkat pada keadaan nyata dimana saat terjadi hubung singkat nilai arus akan semakin besar bila sumber gangguan semakin dekat dengan sumber tegangan atau Gardu Induk.
3. Dapat diketahui juga bahwa saat terjadi arus hubung singkat akan menyebabkan terjadinya *Drop* tegangan.
4. MCB yang digunakan di modul masih memerlukan waktu yang lebih lama untuk *trip* karena MCB yang digunakan dimodul bekerja menggunakan prinsip *thermal* sedangkan pada PLN sudah menggunakan prinsip pengaturan waktu.
5. Selain itu kami waktu pengiriman sms dari SIM 900 rata-rata 8,7 detik juga sangat tergantung dari kondisi sinyal yang ada.

Selain itu dari presentasi alat yang telah dilakukan, alat ini mengalami beberapa kegagalan sistem yakni SMS dari Modem SIM900 yang dikirim ke Modem *Maestro* tidak masuk. Kegagalan tersebut mungkin dikarenakan beberapa faktor yakni:

1. SMS *Pending* akibat buruknya sinyal *provider* Telkomsel yang digunakan.
2. Koneksi kabel yang longgar.
3. Peralatan modem yang kurang baik kondisinya karena yang digunakan adalah peralatan bekas.

5.2 Saran

Beberapa saran yang kami usulkan untuk mengembangkan tugas akhir yakni:

1. Pemilihan *provider Sim Card* sebaiknya diperhatikan dengan baik karena semakin baik *Sim Card* yang digunakan semakin cepat proses pengiriman SMS dapat dilakukan.
2. Alat ini dapat diberi alat ukur pada jaringannya sehingga dapat secara cepat diketahui besarnya arus maupun tegangan yang sedang mengalir.
3. Jenis gangguan dapat ditambah seperti gangguan dua fasa ke tanah dan juga tiga fasa.
4. Menambahkan area kerja alat ini hingga dapat dipasang paada percabangan jaringan distribusi.

DAFTAR PUSTAKA

- [1] Suswanto, D., **Sistem Distribusi Tenaga Listrik**, Teknik Elektro Universitas Negeri Padang, Padang, 2009.
- [2] Wahyudi, SN, **Proteksi Sistem Disribusi Tenaga Listrik**, Garamond, Depok, 2012.
- [3], **Buku 1 Kriteria Disain Enjinering Konstruksi Jaringan Distribusi Tenaga Listrik**, PT PLN (Persero), Jakarta, 2010.
- [4] Chapman, Stephen J., **Electric Machinery Fundamentals**, Fifth Edition, March 2010
- [5] J. Lewis Blackburn, **Protective Relaying Principles and Applications, Second Edition.**
- [6] Burke, J.J. and Lawrence, D. J., "Characteristics of Fault Currents on Distribution Systems," **IEEE Transactions on Power Apparatus and Systems**, vol. PAS-103, no. 1, pp. 1-6, January 1984.
- [7] Murtadho, Moch. Yogi, Prototipe Sistem Pelaporan Gangguan Beserta Posisi Gangguan Pada Jaringan Distribusi, **Tugas Akhir**, Program D3 Teknik Elektro FTI-ITS, Surabaya, 2014.
- [8] Syahwil, M., **Panduan Mudah Simulasi & Praktek Mikrokontroler Arduino**, ANDI, Yogyakarta, 2013.
- [9] Kurniyanto F. W. dan Madina N. F., Purwarupa KWH Meter Digital yang dilengkapi Data Logger sebagai Alat Bantu melakukan Audit Energi, **Tugas Akhir**, Program D3 Teknik Elektro FTI-ITS, Surabaya, 2015.
- [10], **IComSat v1.1-SIM900 GSM/GPRS shield**, Datasheet, Iteadstudio, 2011
- [11], **BelajarDelphi7**, URL: <https://catataneko.wordpress.com/belajar-delphi-7/> (diakses 3 April 2017)

-----*Halaman ini sengaja dikosongkan*-----

LAMPIRAN A

A.1. *Listing Program pada Arduino*

```
#include "Wire.h"
#include <DS3231.h>
#include <LiquidCrystal.h>
#include "EmonLib.h"
#define DS3231_I2C_ADDRESS 0x68//alamat i2c arduino mega
EnergyMonitor emon1;// Create an instance
EnergyMonitor emon2;
EnergyMonitor emon3;
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

byte decToBcd(byte val)// Convert normal decimal numbers to
                        binary coded decimal
{
    return( (val/10*16) + (val%10) );
}
byte bcdToDec(byte val)// Convert binary coded decimal to
                        normal decimal numbers
{
    return( (val/16*10) + (val%16) );
}

//Nomor Khusus
String Nomor ="";
char RcvdMsg[200] = " ";
int RcvdCheck = 0;
int RcvdConf = 0;
int index = 0;
int RcvdEnd = 0;
char NoTelp[15];
char Pesan[50]=" ";
int MsgLength = 0;

//Sensor
float TeganganFasa_R= 0;
float TeganganFasa_S= 0;
float TeganganFasa_T= 0;
```

```

float TeganganFasa_RS= 0;
float TeganganFasa_ST= 0;
float TeganganFasa_3ph= 0;
float ArusFasa_R= 0;
float ArusFasa_S= 0;
float ArusFasa_T= 0;
float ArusFasa_R2= 0;
float ArusFasa_S2= 0;
float ArusFasa_T2= 0;
float Isc_R=0;
float Isc_S=0;
float Isc_T=0;
float Isc_RS=0;
float Isc_ST=0;
int Jarak_R=0;
int Jarak_S=0;
int Jarak_T=0;
int Jarak_RS=0;
int Jarak_ST=0;
boolean Gangguan = false;
String incoming;

//karakter simbol
uint8_t Baik[8]={0x00,0x00,0x01,0x02,0x14,0x08,0x00,0x00};
uint8_t Buruk[8]={0x00,0x11,0x0A,0x04,0x0A,0x11,0x00,0x00};
uint8_t Arus[8]={0x04,0x08,0x10,0x1F,0x01,0x02,0x04,0x08};
uint8_t Jarak1[8]={0x00,0x04,0x08,0x1F,0x08,0x04,0x00,0x00};
uint8_t Jarak2[8]={0x00,0x04,0x02,0x1F,0x02,0x04,0x00,0x00};

void setup()
{
  Nomor.reserve(200);
  lcd.begin(16,2);
  lcd.createChar(1,Baik);
  lcd.createChar(2,Buruk);
  lcd.createChar(3,Arus);
  lcd.createChar(4,Jarak1);
  lcd.createChar(5,Jarak2);
  //kalibrasi sensor arus

```

```

emon1.current(0, 56.18860877684407); // Current: input pin,
                                     nilai kalibrasi (fasa S).
emon2.current(1, 56.30065352);      // Current: input pin, nilai
                                     kalibrasi(fasa R).
emon3.current(2, 56.30065352);      // Current: input pin, nilai
                                     kalibrasi(fasa T).

//power SIM900
pinMode(9,OUTPUT);

//indikator
pinMode(14,OUTPUT); //R
pinMode(15,OUTPUT); //S
pinMode(16,OUTPUT); //T
digitalWrite(14,HIGH); //R
digitalWrite(15,HIGH); //S
digitalWrite(16,HIGH); //T
SIM900power();

//tampilan awal
lcd.setCursor(0,0);
lcd.print(" Bismillah");
lcd.setCursor(0,1);
lcd.print(" TUGAS AKHIR ");
delay(1500);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Prototype Sistem");
lcd.setCursor(0,1);
lcd.print("Deteksi Gangguan");

Wire.begin();
//set RTC mula2:
//DS3231 seconds, minutes, hours, dayOfMonth, month, year;
//setDS3231time(00,46,1,15,7,17);
Serial.begin(9600);
Serial1.begin(9600);
delay(1000);
Serial1.print("AT+CNMI=2,2,0,0,0\r");
Serial1.println("AT+CMGF=1\r\n");

```

```

delay(3000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" R:");
lcd.write(1);
lcd.print(" S:");
lcd.write(1);
lcd.print(" T:");
lcd.write(1);
delay(1000);
}

```

```

void(* resetFunc) (void) = 0;//declare reset function at address 0

```

```

void loop()
{
  MenerimaSMS();

  while ( Gangguan == false)
  {
    double Irms = emon1.calcIrms(1480); // Calculate Irms only(S)
    double Irms2 = emon2.calcIrms(1480); // CalculateIrms only(R)
    double Irms3 = emon3.calcIrms(1480); // CalculateIrms only(T)
    ArusFasa_S= Irms;
    ArusFasa_R= Irms2;
    ArusFasa_T= Irms3;
    if(ArusFasa_R>=13){ ArusFasa_R=0;}
    if(ArusFasa_S>=13){ ArusFasa_S=0;}
    if(ArusFasa_T>=13){ ArusFasa_T=0;}
    TeganganFasa_R= (((analogRead(A7)*0.014)+0.6432)-1.2);
    TeganganFasa_S= (((analogRead(A6)*0.014)+0.6432)-1.65);
    TeganganFasa_T= (((analogRead(A5)*0.0142)+0.629)-1.5);
    if(TeganganFasa_R==0.6432)
    {
      TeganganFasa_R=0;
    }
    if(TeganganFasa_S==0.6432)
    {
      TeganganFasa_S=0;
    }
  }
}

```

```

    }
    if(TeganganFasa_T==0.629)
    {
        TeganganFasa_T=0;
    }

    if( ArusFasa_R > 4.78 and ArusFasa_R < 4.95)
    {
        ArusFasa_R=0;
    }
    if( ArusFasa_S > 4.78 and ArusFasa_S < 4.95)
    {
        ArusFasa_S=0;
    }
    if( ArusFasa_T > 4.78 and ArusFasa_T < 4.95)
    {
        ArusFasa_T=0;
    }

    displayTimeLCDNormal();
    if (ArusFasa_R < ArusFasa_R2)
    {
        if (ArusFasa_S < ArusFasa_S2 and ArusFasa_S2 < 5.2 and
        ArusFasa_R2 < 5.2)
        {
            Isc_RS= (ArusFasa_R2+ArusFasa_S2)/2;
            TeganganFasa_RS =(TeganganFasa_R+TeganganFasa_S)/2;
            Jarak_RS=(TeganganFasa_RS/Isc_RS)/0.33;
            if (Isc_RS >= 4)
            {
                TampilLCD_RS();
            }
        }
        else
        {
            ArusFasa_R2=ArusFasa_R;
            ArusFasa_S2=ArusFasa_S;
        }
    }
    else if (ArusFasa_R2 > 5.5)

```

```

{
  Isc_R=ArusFasa_R2;
  Jarak_R=(TeganganFasa_R/Isc_R)/0.33;
  if (Isc_R >= 5)
  {
    TampilLCD_R();
  }
  else
  {
    ArusFasa_R2=ArusFasa_R;
  }
}
}
else
{
  ArusFasa_R2=ArusFasa_R;
}

if (ArusFasa_S < ArusFasa_S2)
{
  if (ArusFasa_T < ArusFasa_T2 and ArusFasa_T2 > 4.7 and
  ArusFasa_S2 < 6.5)
  {
    Isc_ST= (ArusFasa_T2+ArusFasa_S2)/2;
    TeganganFasa_ST =(TeganganFasa_S+TeganganFasa_T)/2;
    Jarak_ST=(TeganganFasa_ST/Isc_ST)/0.33;
    if (Isc_ST >= 4)
    {
      TampilLCD_ST();
    }
    else
    {
      ArusFasa_T2=ArusFasa_T;
      ArusFasa_S2=ArusFasa_S;
    }
  }
  else if (ArusFasa_S2 > 7)
  {
    Isc_S=ArusFasa_S2;
  }
}

```



```

    Jarak_S=(TeganganFasa_S/Isc_S)/0.33;
    if (Isc_S >= 4.9)
    {
        TampilLCD_S();
    }
    else
    {
        ArusFasa_S2=ArusFasa_S;
    }
}
else
{
    return Isc_RS;
}
}
else
{
    ArusFasa_S2=ArusFasa_S;
}

if (ArusFasa_T < ArusFasa_T2 and ArusFasa_T2 <4.5)
{
    Isc_T=ArusFasa_T2;
    Jarak_T=(TeganganFasa_T/Isc_T)/0.33;
    if (Isc_T >= 3.5)
    {
        TampilLCD_T();
    }
    else
    {
        ArusFasa_T2=ArusFasa_T;
    }
}
else
{
    ArusFasa_T2=ArusFasa_T;
}
} //closing fungsi while
} // closing void loop

```

```

void SIM900power()
{
    digitalWrite(9,HIGH);
    delay(500);
    digitalWrite(9,LOW);
    return;
}

```

```

void TampilLCD_R()
{
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" R:");
    lcd.write(2);
    lcd.print(" S:");
    lcd.write(1);
    lcd.print(" T:");
    lcd.write(1);
    //lcd.clear();
    lcd.setCursor(0,1);
    lcd.write(3);
    lcd.print(":");
    lcd.print(Isc_R);
    lcd.print(" A ");
    lcd.write(4);
    lcd.write(5);
    lcd.print(":");
    lcd.print(Jarak_R);
    lcd.print("KM");
    digitalWrite(14,LOW);
    digitalWrite(15,HIGH);
    digitalWrite(16,HIGH);
    Gangguan = true;
    SMS_R();
    delay(3000);
    SMS_R_PC();
}

```

```

void TampilLCD_S()
{
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" R:");
    lcd.write(1);
    lcd.print(" S:");
    lcd.write(2);
    lcd.print(" T:");
    lcd.write(1);
    lcd.setCursor(0,1);
    lcd.write(3);
    lcd.print(":");
    lcd.print(Isc_S);
    lcd.print("A ");
    lcd.write(4);
    lcd.write(5);
    lcd.print(":");
    lcd.print(Jarak_S);
    lcd.print("KM");
    digitalWrite(14,HIGH);
    digitalWrite(15,LOW);
    digitalWrite(16,HIGH);
    Gangguan = true;
    SMS_S();
    delay(3000);
    SMS_S_PC();
}

```

```

void TampilLCD_T()
{
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" R:");
    lcd.write(1);
    lcd.print(" S:");
    lcd.write(1);
    lcd.print(" T:");
}

```

```

lcd.write(2);
lcd.setCursor(0,1);
lcd.write(3);
lcd.print(":");
lcd.print(Isc_T);
lcd.print("A ");
lcd.write(4);
lcd.write(5);
lcd.print(":");
lcd.print(Jarak_T);
lcd.print("KM");
digitalWrite(14,HIGH);
digitalWrite(15,HIGH);
digitalWrite(16,LOW);
Gangguan = true;
SMS_T();
delay(3000);
SMS_T_PC();
}

```

```

void TampilLCD_RS()
{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" R:");
  lcd.write(2);
  lcd.print(" S:");
  lcd.write(2);
  lcd.print(" T:");
  lcd.write(1);
  lcd.setCursor(0,1);
  lcd.write(3);
  lcd.print(":");
  lcd.print(Isc_RS);
  lcd.print("A ");
  lcd.write(4);
  lcd.write(5);
  lcd.print(":");
  lcd.print(Jarak_RS);
}

```

```

    lcd.print("KM");
    digitalWrite(14,LOW);
    digitalWrite(15,LOW);
    digitalWrite(16,HIGH);
    Gangguan = true;
    SMS_RS();
    delay(3000);
    SMS_RS_PC();
}

```

```

void TampilLCD_ST()
{
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" R:");
    lcd.write(1);
    lcd.print(" S:");
    lcd.write(2);
    lcd.print(" T:");
    lcd.write(2);
    lcd.setCursor(0,1);
    lcd.write(3);
    lcd.print(":");
    lcd.print(Isc_ST);
    lcd.print("A ");
    lcd.write(4);
    lcd.write(5);
    lcd.print(":");
    lcd.print(Jarak_ST);
    lcd.print("KM");
    digitalWrite(14,HIGH);
    digitalWrite(15,LOW);
    digitalWrite(16,LOW);
    Gangguan = true;
    SMS_ST();
    delay(3000);
    SMS_ST_PC();
}

```

```

void SMS_R()
{
    Serial1.print("AT+CMGS=\"");
    Serial1.print("+6281330087465");
    Serial1.print("\r\n");
    delay(1000);
    Serial1.println("Gangguan satu fasa ke tanah");
    Serial1.println("Fasa R");
    Serial1.print("Jarak = ");
    Serial1.print(Jarak_R);
    Serial1.println(" KM");
    Serial1.print("Arus = ");
    Serial1.print(Isc_R);
    Serial1.println(" A");
    Serial1.print("Tegangan = ");
    Serial1.print(TeganganFasa_R);
    Serial1.println(" V");
    displayTimesms();
    Serial1.println("");
    Serial1.write(0x1A);//akhir dari pesan
    Gangguan = true;
}

```

```

void SMS_S()
{
    Serial1.print("AT+CMGS=\"");
    Serial1.print("+6281330087465");
    Serial1.print("\r\n");
    delay(1000);
    Serial1.println("Gangguan satu fasa ke tanah");
    Serial1.println("Fasa S");
    Serial1.print("Jarak = ");
    Serial1.print(Jarak_S);
    Serial1.println(" KM");
    Serial1.print("Arus = ");
    Serial1.print(Isc_S);
    Serial1.println(" A");
    Serial1.print("Tegangan = ");
}

```

```

Serial1.print(TeganganFasa_S);
Serial1.println(" V");
displayTimesms();
Serial1.println("");
Serial1.write(0x1A);//akhir dari pesan
Gangguan = true;
}

void SMS_T()
{
  Serial1.print("AT+CMGS=\"");
  Serial1.print("+6281330087465");
  Serial1.print("\\r\\n");
  delay(1000);
  Serial1.println("Gangguan satu fasa ke tanah");
  Serial1.println("Fasa T");
  Serial1.print("Jarak =");
  Serial1.print(Jarak_T);
  Serial1.println(" KM");
  Serial1.print("Arus = ");
  Serial1.print(Isc_T);
  Serial1.println(" A");
  Serial1.print("Tegangan = ");
  Serial1.print(TeganganFasa_T);
  Serial1.println(" V");
  displayTimesms();
  Serial1.println("");
  Serial1.write(0x1A);//akhir dari pesan
  Gangguan = true;
}

void SMS_RS()
{
  Serial1.print("AT+CMGS=\"");
  Serial1.print("+6281330087465");
  Serial1.print("\\r\\n");
  delay(1000);
  Serial1.println("Gangguan antar fasa");
  Serial1.println("Fasa R dan S");
}

```

```

Serial1.print("Jarak =");
Serial1.print(Jarak_RS);
Serial1.println(" KM");
Serial1.print("Arus = ");
Serial1.print(Isc_RS);
Serial1.println(" A");
Serial1.print("Tegangan = ");
Serial1.print(TeganganFasa_RS);
Serial1.println(" V");
displayTimesms();
Serial1.println("");
Serial1.write(0x1A);//akhir dari pesan
Gangguan = true;
}

void SMS_ST()
{
  Serial1.print("AT+CMGS=\"");
  Serial1.print("+6281330087465");
  Serial1.print("\r\n");
  delay(1000);
  Serial1.println("Gangguan antar fasa");
  Serial1.println("Fasa S dan T");
  Serial1.print("Jarak =");
  Serial1.print(Jarak_ST);
  Serial1.println(" KM");
  Serial1.print("Arus = ");
  Serial1.print(Isc_ST);
  Serial1.println(" A");
  Serial1.print("Tegangan = ");
  Serial1.print(TeganganFasa_ST);
  Serial1.println(" V");
  displayTimesms();
  Serial1.println("");
  Serial1.write(0x1A);//akhir dari pesan
  Gangguan = true;
}

```



```

void SMS_R_PC()
{
    Serial1.print("AT+CMGS=\"");
    Serial1.print("+6282132318298");
    Serial1.print("\r\n");
    delay(1000);
    Serial1.println("Gangguan satu fasa ke tanah/");
    Serial1.println("Fasa R/");
    Serial1.print("Jarak =");
    Serial1.print(Jarak_R);
    Serial1.println(" KM/");
    Serial1.print("Arus = ");
    Serial1.print(Isc_R);
    Serial1.println(" A/");
    Serial1.print("Tegangan = ");
    Serial1.print(TeganganFasa_R);
    Serial1.println(" V");
    displayTimesms();
    Serial1.println("");
    Serial1.write(0x1A); //akhir dari pesan
}

```

```

void SMS_S_PC()
{
    Serial1.print("AT+CMGS=\"");
    Serial1.print("+6282132318298");
    Serial1.print("\r\n");
    delay(1000);
    Serial1.println("Gangguan satu fasa ke tanah/");
    Serial1.println("Fasa S/");
    Serial1.print("Jarak =");
    Serial1.print(Jarak_S);
    Serial1.println(" KM/");
    Serial1.print("Arus = ");
    Serial1.print(Isc_S);
    Serial1.println(" A/");
    Serial1.print("Tegangan = ");
    Serial1.print(TeganganFasa_S);
}

```

```

Serial1.println(" V");
displayTimesms();
Serial1.println("");
Serial1.write(0x1A);//akhir dari pesan
}

void SMS_T_PC()
{
  Serial1.print("AT+CMGS=\"");
  Serial1.print("+6282132318298");
  Serial1.print("\r\n");
  delay(1000);
  Serial1.println("Gangguan satu fasa ke tanah/");
  Serial1.println("Fasa T/");
  Serial1.print("Jarak =");
  Serial1.print(Jarak_T);
  Serial1.println(" KM/");
  Serial1.print("Arus = ");
  Serial1.print(Isc_T);
  Serial1.println(" A/");
  Serial1.print("Tegangan = ");
  Serial1.print(TeganganFasa_T);
  Serial1.println(" V");
  displayTimesms();
  Serial1.println("");
  Serial1.write(0x1A);//akhir dari pesan
}

void SMS_RS_PC()
{
  Serial1.print("AT+CMGS=\"");
  Serial1.print("+6282132318298");
  Serial1.print("\r\n");
  delay(1000);
  Serial1.println("Gangguan antar fasa/");
  Serial1.println("Fasa R dan S/");
  Serial1.print("Jarak =");
  Serial1.print(Jarak_RS);
  Serial1.println(" KM/");
}

```

```

Serial1.print("Arus = ");
Serial1.print(Isc_RS);
Serial1.println(" A");
Serial1.print("Tegangan = ");
Serial1.print(TeganganFasa_RS);
Serial1.println(" V");
displayTimesms();
Serial1.println("");
Serial1.write(0x1A);//akhir dari pesan
}

```

```

void SMS_ST_PC()
{
    Serial1.print("AT+CMGS=\"");
    Serial1.print("+6282132318298");
    Serial1.print("\r\n");
    delay(1000);
    Serial1.println("Gangguan antar fasa");
    Serial1.println("Fasa S dan T");
    Serial1.print("Jarak =");
    Serial1.print(Jarak_ST);
    Serial1.println(" KM");
    Serial1.print("Arus = ");
    Serial1.print(Isc_ST);
    Serial1.println(" A");
    Serial1.print("Tegangan = ");
    Serial1.print(TeganganFasa_ST);
    Serial1.println(" V");
    displayTimesms();
    Serial1.println("");
    Serial1.write(0x1A);//akhir dari pesan
}

```

```

void setDS3231time(byte second, byte minute, byte hour, byte
dayOfMonth, byte month, byte year)
{
    // sets time and date data to DS3231
    Wire.beginTransaction(DS3231_I2C_ADDRESS);
    Wire.write(0); // set next input to start at the seconds register
}

```

```

Wire.write(decToBcd(second)); // set seconds
Wire.write(decToBcd(minute)); // set minutes
Wire.write(decToBcd(hour)); // set hours
Wire.write(decToBcd(dayOfMonth)); // set date (1 to 31)
Wire.write(decToBcd(month)); // set month
Wire.write(decToBcd(year)); // set year (0 to 99)
Wire.endTransmission();
}

void readDS3231time(byte *second,byte *minute,byte *hour,byte
*dayOfMonth,byte *month,byte *year)
{
Wire.beginTransmission(DS3231_I2C_ADDRESS);
Wire.write(0); // set DS3231 register pointer to 00h
Wire.endTransmission();
Wire.requestFrom(DS3231_I2C_ADDRESS, 7);
// request seven bytes of data from DS3231 starting from register
00h
*second = bcdToDec(Wire.read() & 0x7f);
*minute = bcdToDec(Wire.read());
*hour = bcdToDec(Wire.read() & 0x3f);
*dayOfMonth = bcdToDec(Wire.read());
*month = bcdToDec(Wire.read());
*year = bcdToDec(Wire.read());
}

void displayTimeLCDNormal()
{
byte second, minute, hour, dayOfMonth, month, year;
// retrieve data from DS3231
readDS3231time(&second, &minute, &hour, &dayOfMonth,
&month, &year);
lcd.setCursor(0,1);
lcd.print(dayOfMonth, DEC);//merubah variabel byte ke angka
desimal ketika ditampilkan
lcd.print("/");
lcd.print(month, DEC);
lcd.print("/");
lcd.print(year, DEC);
}

```

```

lcd.print(" ");
lcd.setCursor(8,1);
if (hour<10)
{
  lcd.print("0");
}
lcd.print(hour, DEC);
lcd.print(":");
if (minute<10)
{
  lcd.print("0");
}
lcd.print(minute, DEC);
lcd.print(":");
if (second<10)
{
  lcd.print("0");
}
lcd.print(second, DEC);
delay(200);
}

```

```

void displayTimesms()
{
  byte second, minute, hour, dayOfMonth, month, year; // retrieve
  data from DS3231
  readDS3231time(&second, &minute, &hour, &dayOfMonth,
  &month, &year); // send it to the serial monitor
  Serial1.print("[");
  Serial1.print(dayOfMonth, DEC); // convert the byte variable to a
  decimal number when displayed
  Serial1.print("/");
  Serial1.print(month, DEC);
  Serial1.print("/");
  Serial1.print(year, DEC);
  Serial1.print("]");
  Serial1.print(" ");
  Serial1.print("[");
  Serial1.print(hour, DEC);

```

```

Serial1.print(":");
if (minute<10)
{
    Serial1.print("0");
}
Serial1.print(minute, DEC);
Serial1.print(":");
if (second<10)
{
    Serial1.print("0");
}
Serial1.print(second, DEC);
Serial1.println("]");
}

```

```

void MenerimaSMS()
{
    if(Serial1.available()>0)
    {
        char data = Serial1.read();
        if(data == '+'){RcvdCheck = 1;}
        if((data == 'C') && (RcvdCheck == 1)){RcvdCheck = 2;}
        if((data == 'M') && (RcvdCheck == 2)){RcvdCheck = 3;}
        if((data == 'T') && (RcvdCheck == 3)){RcvdCheck = 4;}
        if(RcvdCheck == 4){index = 0; RcvdConf = 1; RcvdCheck = 0;}
        if(RcvdConf == 1)
        {
            if(data == '\n'){RcvdEnd++;}
            if(RcvdEnd == 3){RcvdEnd = 0;}
            RcvdMsg[index] = data;
            index++;
            if(RcvdEnd == 2){RcvdConf = 0;MsgLength = index-2;}
            if(RcvdConf == 0)
            {
                for(int x = 4;x <18;x++)
                {
                    NoTelp[x-4] = RcvdMsg[x];
                    Nomor += NoTelp[x-4];
                }
            }
        }
    }
}

```

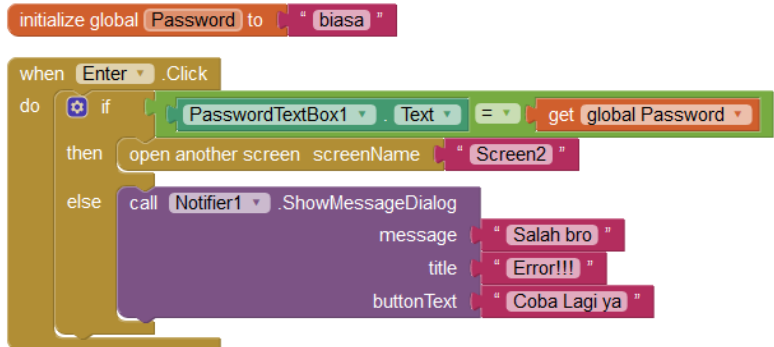
```

for(int x = 47;x <MsgLength;x++)
{
    Pesan[x-47] = RcvdMsg[x];
}
if (Nomor=="6281330087465")
{
    if(Pesan[0]=='#')
    {
        resetFunc();
    }
}
Nomor="";
strcpy(RcvdMsg,"");
RcvdCheck = 0;
RcvdConf = 0;
index = 0;
RcvdEnd = 0;
strcpy(NoTelp,"");
strcpy(Pesan,"");
MsgLength = 0;
}
}
}
}

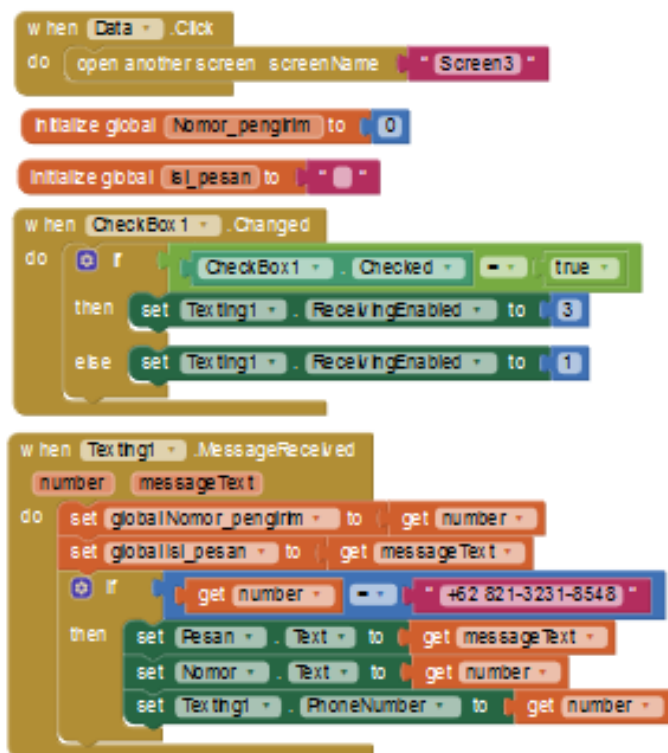
```

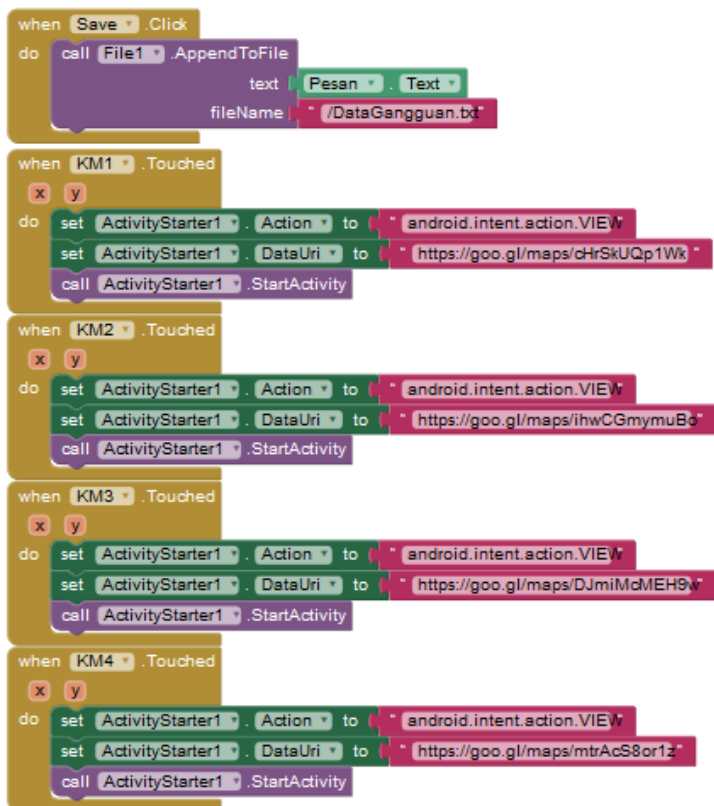
A.2. Listing Block Editor MIT App Inventor

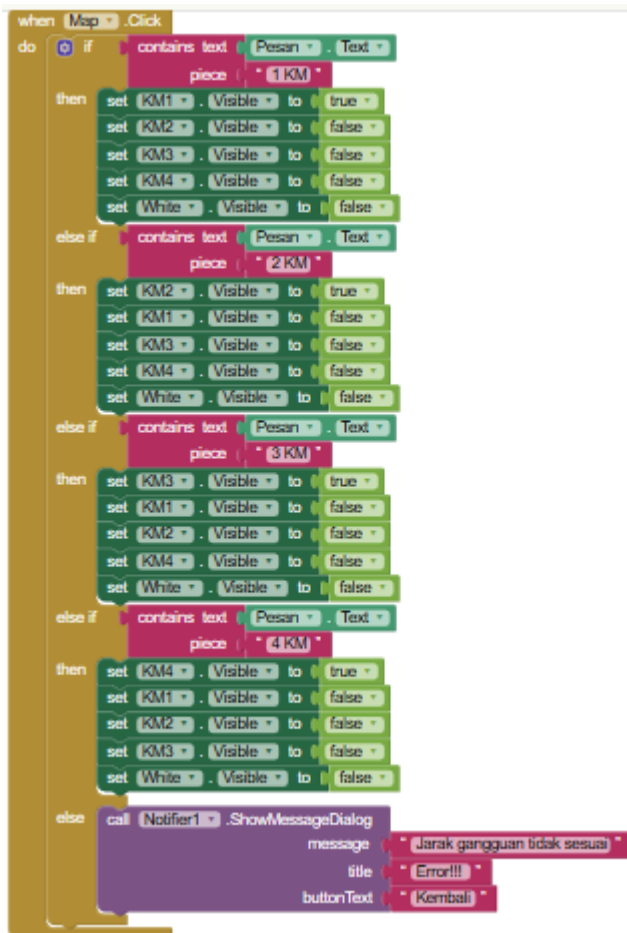
1. Program Block Screen 1



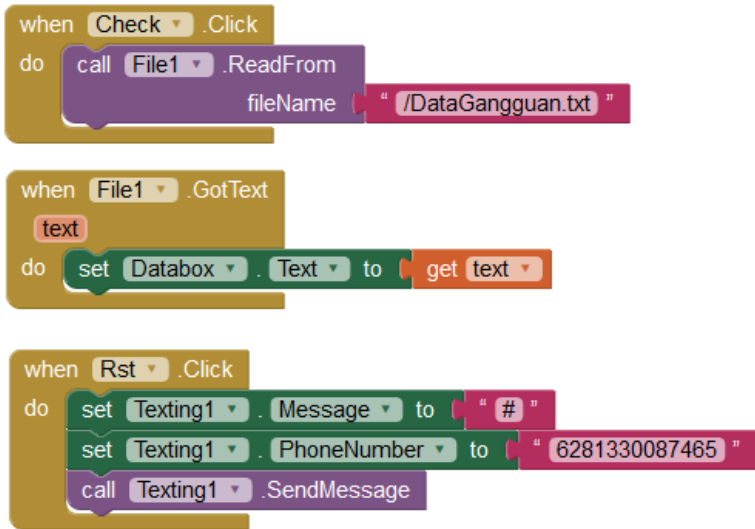
2. Program Block *Screen 2*







3. Program Block *Screen 3*



A.3. Listing Program Pada Delphi 7

```
unit Unit1;
```

```
interface
```

```
uses
```

```
  Windows, Messages, SysUtils, Variants, Classes, Graphics,  
  Controls, Forms,  
  Dialogs, CPort, StdCtrls, StrUtils, DB, ExtCtrls, DBCtrls, Grids,  
  DBGrids, ADODB;
```

```
type
```

```
  TForm1 = class(TForm)  
    Edit1: TEdit;  
    Memo1: TMemo;  
    Edit2: TEdit;  
    Edit3: TEdit;
```

```

Edit4: TEdit;
Edit5: TEdit;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
ComPort1: TComPort;
Button1: TButton;
Button2: TButton;
Button3: TButton;
Button4: TButton;
ADOConnection1: TADOConnection;
ADOTable1: TADOTable;
DBGrid1: TDBGrid;
DBNavigator1: TDBNavigator;
DataSource1: TDataSource;
gmbrPLN: TImage;
Label7: TLabel;
Label8: TLabel;
its: TImage;
Shape1: TShape;
GroupBox1: TGroupBox;
GroupBox2: TGroupBox;
Label9: TLabel;
Label10: TLabel;
Label11: TLabel;
Label12: TLabel;
Shape4: TShape;
Label13: TLabel;
Shape2: TShape;
Label14: TLabel;
ReTime: TTimer;
procedure varRxChar(Sender: TObject; Count: Integer);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure Button4Click(Sender: TObject);

```

```

procedure Button3Click(Sender: TObject);
procedure ReTimeTimer(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.varRxChar(Sender: TObject; Count: Integer);
var
  data: String;
  kar_awal,kar_akhir:integer;
begin
  ComPort1.ReadStr(data, 300);
  Memo1.Text := Memo1.Text + data;
  begin
    kar_awal:=pos ('Gangguan',Memo1.Text);
    kar_akhir:=posex ('/',Memo1.Text,kar_awal+1);
    edit3.Text :=copy(Memo1.Text,kar_awal,kar_akhir-kar_awal);
    kar_awal:=posex ('Fasa',Memo1.Text,kar_akhir+1);
    kar_akhir:=posex ('/',Memo1.Text,kar_awal+1);
    edit4.Text :=copy(Memo1.Text,kar_awal,kar_akhir-kar_awal);
    kar_awal:=posex ('=',Memo1.Text,kar_akhir+1);
    kar_akhir:=posex ('/',Memo1.Text,kar_awal+1);
    edit5.Text:=copy(Memo1.Text,kar_awal+1,kar_akhir-kar_awal-1);
    kar_awal:=posex ('(',Memo1.Text,kar_akhir+1);
    kar_akhir:=posex (')',Memo1.Text,kar_awal+1);
    edit1.Text:=copy(Memo1.Text,kar_awal+1,kar_akhir-kar_awal-1);
    kar_awal:=posex ('(',Memo1.Text,kar_akhir+1);
    kar_akhir:=posex (')',Memo1.Text,kar_awal+1);
    edit2.Text:=copy(Memo1.Text,kar_awal+1,kar_akhir-kar_awal-1);
  end;
end;

```

```

end;

begin
  if edit4.Text='Fasa R' then
    begin
      Shape1.brush.color:=clBlack;
    end
  else if edit4.Text='Fasa S' then
    begin
      Shape4.brush.color:=clBlack;
    end
  else if edit4.Text='Fasa T' then
    begin
      Shape2.brush.color:=clBlack;
    end
  else if edit4.Text='Fasa R dan S' then
    begin
      Shape1.brush.color:=clBlack;
      Shape4.brush.color:=clBlack;
    end
  else if edit4.Text='Fasa S dan T' then
    begin
      Shape4.brush.color:=clBlack;
      Shape2.brush.color:=clBlack;
    end;
  end;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  comport1.ShowSetupDialog;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
  if ComPort1.Connected then
    begin
      ComPort1.Close;
      button2.Caption:='Open';
    end
  end
end

```

```

else
begin
    ComPort1.Open;
    button2.Caption:='Close';
    ComPort1.WriteStr('AT+CMGF=1'+#13#10);
    ComPort1.WriteStr('AT+CPMS="SM"+#13#10);
    ComPort1.WriteStr('AT+CSCS="GSM"+#13#10);
end;
end;

procedure TForm1.FormCreate(Sender: TObject);
var alamatdb : WideString;
begin
    alamatdb := ExtractFilePath(Application.ExeName) + 'datasms.mdb';
    with ADOConnection1 do
    begin
        Connected := False;
        LoginPrompt := False;
        Mode := cmShareDenyNone;
        ConnectionString := 'Provider=Microsoft.Jet.OLEDB.4.0;Data
        Source=' +
        alamatdb + ';Persist Security Info=False';
    end;
    ADOConnection1.Connected := True;
    ADOtable1.Connection:= ADOConnection1;
    ADOtable1.TableName:='datasms';
    ADOtable1.Active:=true;
    Datasource1.DataSet:= ADOTable1;
    DBGrid1.DataSource:= DataSource1;

    edit1.Text:="";
    edit2.Text:="";
    edit3.Text:="";
    edit4.Text:="";
    edit5.Text:="";
    Memo1.Text:="";
end;

```

```

procedure TForm1.Button4Click(Sender: TObject);
begin
    adotable1.Open;
    adotable1.Append;
    adotable1.FieldByName('Tanggal').AsString:=edit1.text;
    adotable1.FieldByName('Waktu').AsString:=edit2.text;
    adotable1.FieldByName('Jenis Gangguan').AsString:=edit3.text;
    adotable1.FieldByName('Fasa Yang Terganggu').AsString:=
edit4.text;
    adotable1.FieldByName('Jarak Gangguan').AsString:=edit5.text;
    adotable1.post;
    begin
        memo1.Clear;
        comport1.WriteStr('at+cmgd=1,4'+#13#10);
    end;
    begin
        Shape4.brush.color:=clYellow;
        Shape2.brush.color:=clLime;
        Shape1.brush.color:=clRed;
    end;
end;

```

```

procedure TForm1.Button3Click(Sender: TObject);
begin
    edit1.Text:="";
    edit2.Text:="";
    edit3.Text:="";
    edit4.Text:="";
    edit5.Text:="";
    Memo1.Text:="";
end;

```

```

procedure TForm1.ReTimeTimer(Sender: TObject);
begin
    Memo1.Clear;
    ComPort1.WriteStr('at+cmgr="1"+#13#10);
end;
end.

```


LAMPIRAN B

B.1 DATASHEET RTC DS3231

DS3231

Extremely Accurate I²C-Integrated RTC/TCXO/Crystal

General Description

The DS3231 is a low-cost, extremely accurate I²C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The DS3231 is available in commercial and industrial temperature ranges, and is offered in a 16-pin, 300-mil SO package.

The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM Indicator. Two programmable time-of-day alarms and a programmable square-wave output are provided. Address and data are transferred serially through an I²C bidirectional bus.

A precision temperature-compensated voltage reference and comparator circuit monitors the status of V_{CC} to detect power failures, to provide a reset output, and to automatically switch to the backup supply when necessary. Additionally, the RST pin is monitored as a pushbutton input for generating a µP reset.

Benefits and Features

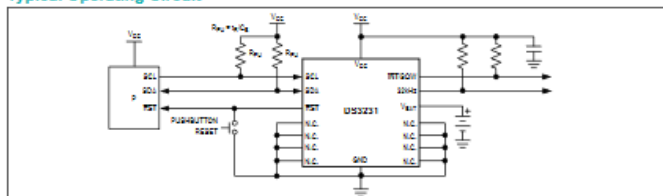
- Highly Accurate RTC Completely Manages All Timekeeping Functions
 - Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year, with Leap-Year Compensation Valid Up to 2100
 - Accuracy ±2ppm from 0°C to +40°C
 - Accuracy ±3.5ppm from -40°C to +85°C
 - Digital Temp Sensor Output: ±3°C Accuracy
 - Register for Aging Trim
 - RST Output/Pushbutton Reset Debounce Input
 - Two Time-of-Day Alarms
 - Programmable Square-Wave Output Signal
- Simple Serial Interface Connects to Most Microcontrollers
 - Fast (400KHz) I²C Interface
- Battery-Backup Input for Continuous Timekeeping
 - Low Power Operation Extends Battery-Backup Run Time
 - 3.3V Operation
- Operating Temperature Ranges: Commercial (0°C to +70°C) and Industrial (-40°C to +85°C)
- Underwriters Laboratories® (UL) Recognized

Applications

- Servers
 - Utility Power Meters
- Telematics
 - GPS

Ordering Information and Pin Configuration appear at end of data sheet.

Typical Operating Circuit



Underwriters Laboratories is a registered certification mark of Underwriters Laboratories Inc.

19-0170, Rev 10, 2/15



Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground-0.3V to +6.0V	Junction Temperature+125°C
Junction-to-Ambient Thermal Resistance (R_{JA}) (Note 1)	173°C/W	Storage Temperature Range-40°C to +85°C
Junction-to-Case Thermal Resistance (R_{JC}) (Note 1)	23°C/W	Lead Temperature (soldering, 10s)+260°C
Operating Temperature Range		Soldering Temperature (reflow, 2 times max)+260°C
DS3231S0°C to +70°C	(see the Handling, PCB Layout, and Assembly section)	
DS3231SN-40°C to +85°C		

Note 1: Package thermal resistances were obtained using the method described in JEDEC specification JEDEC51-7, using a four-layer board. For detailed information on package thermal considerations, refer to www.maximintegrated.com/thermal-tutorial.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

($T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Supply Voltage	V_{CC}		2.3	3.3	5.5	V
	V_{BAT}		2.3	3.0	5.5	V
Logic 1 Input SDA, SCL	V_{IH}		0.7 x V_{CC}		$V_{CC} + 0.3$	V
Logic 0 Input SDA, SCL	V_{IL}		-0.3		0.3 x V_{CC}	V

Electrical Characteristics

($V_{CC} = 2.3V$ to $5.5V$, V_{CC} = Active Supply (see Table 1), $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Typical values are at $V_{CC} = 3.3V$, $V_{BAT} = 3.0V$, and $T_A = +25°C$, unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Active Supply Current	I_{CCA}	(Notes 4, 5)	$V_{CC} = 3.63V$		200	μA
			$V_{CC} = 5.5V$		300	
Standby Supply Current	I_{CCS}	I2C bus inactive, 32kHz output on, 8QW output off (Note 5)	$V_{CC} = 3.63V$		110	μA
			$V_{CC} = 5.5V$		170	
Temperature Conversion Current	$I_{CCSCONV}$	I2C bus inactive, 32kHz output on, 8QW output off	$V_{CC} = 3.63V$		575	μA
			$V_{CC} = 5.5V$		660	
Power-Fail Voltage	V_{PF}		2.45	2.575	2.70	V
Logic 0 Output, 32kHz, INT/SQW, SDA	V_{OL}	$I_{OL} = 3mA$			0.4	V
Logic 0 Output, RST	V_{OL}	$I_{OL} = 1mA$			0.4	V
Output Leakage Current 32kHz, INT/SQW, SDA	I_{LO}	Output high impedance	-1	0	+1	μA
Input Leakage SCL	I_{LI}		-1		+1	μA
RST Pin I/O Leakage	I_{OL}	RST high impedance (Note 6)	-200		+10	μA
V_{BAT} Leakage Current (V_{CC} Active)	I_{BATLKG}			25	100	nA

Electrical Characteristics (continued)

(V_{CC} = 2.3V to 5.5V, V_{CC} = Active Supply (see Table 1), T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Typical values are at V_{CC} = 3.3V, V_{BAT} = 3.0V, and T_A = +25°C, unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Output Frequency	f _{OUT}	V _{CC} = 3.3V or V _{BAT} = 3.3V		32.768		kHz
Frequency Stability vs. Temperature (Commercial)	Δf _{f_{OUT}}	V _{CC} = 3.3V or V _{BAT} = 3.3V, aging offset = 00h	0°C to +40°C	±2		ppm
			+40°C to +70°C	±3.5		
Frequency Stability vs. Temperature (Industrial)	Δf _{f_{OUT}}	V _{CC} = 3.3V or V _{BAT} = 3.3V, aging offset = 00h	-40°C to +40°C	±3.5		ppm
			0°C to +40°C	±2		
			+40°C to +85°C	±3.5		
Frequency Stability vs. Voltage	Δf/V			1		ppm/V
Trim Register Frequency Sensitivity per LSB	Δf/LSB	Specified at:	-40°C	0.7		ppm
			+25°C	0.1		
			+70°C	0.4		
			+85°C	0.8		
Temperature Accuracy	Temp	V _{CC} = 3.3V or V _{BAT} = 3.3V		-3	+3	°C
Crystal Aging	Δf _{f_{OUT}}	After reflow, not production tested	First year	±1.0		ppm
			0–10 years	±5.0		

Electrical Characteristics

(V_{CC} = 0V, V_{BAT} = 2.5V to 5.5V, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Active Battery Current	I _{BATA}	EOSC = 0, BB8QW = 0, SCL = 400kHz (Note 5)	V _{BAT} = 3.63V		70	μA
			V _{BAT} = 5.5V		150	
Timekeeping Battery Current	I _{BATT}	EOSC = 0, BB8QW = 0, EN32kHz = 1, SCL = SDA = DV or SCL = SDA = V _{BAT} (Note 5)	V _{BAT} = 3.63V	0.84	3.0	μA
			V _{BAT} = 5.5V	1.0	3.5	
Temperature Conversion Current	I _{BATTC}	EOSC = 0, BB8QW = 0, SCL = SDA = DV or SCL = SDA = V _{BAT}	V _{BAT} = 3.63V		575	μA
			V _{BAT} = 5.5V		650	
Data-Retention Current	I _{BATTOR}	EOSC = 1, SCL = SDA = DV, +25°C			100	nA

AC Electrical Characteristics

(V_{CC} = V_{CC(MIN)} to V_{CC(MAX)} or V_{BAT} = V_{BAT(MIN)} to V_{BAT(MAX)}, V_{BAT} > V_{CC}; T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Note 2)

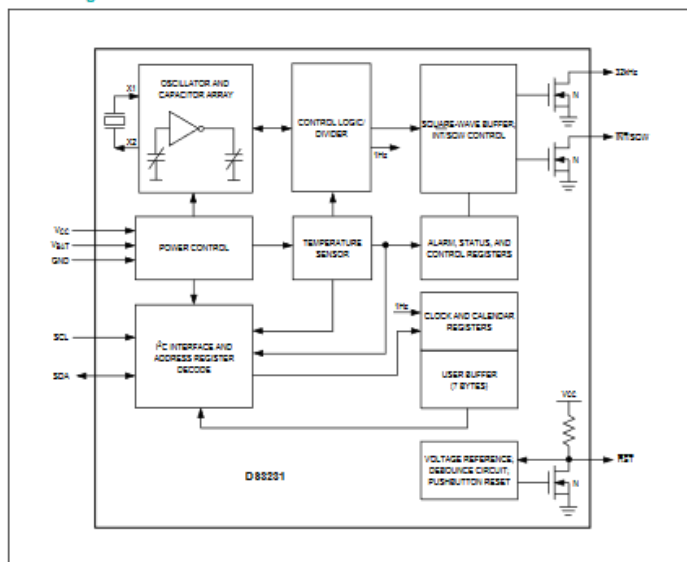
PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f _{SCL}	Fast mode	100		400	kHz
		Standard mode	0		100	
Bus Free Time Between STOP and START Conditions	t _{SUP}	Fast mode	1.3			μs
		Standard mode	4.7			
Hold Time (Repeated) START Condition (Note 7)	t _{HO,STA}	Fast mode	0.6			μs
		Standard mode	4.0			
Low Period of SCL Clock	t _{LOW}	Fast mode	1.3			μs
		Standard mode	4.7			
High Period of SCL Clock	t _{HIGH}	Fast mode	0.6			μs
		Standard mode	4.0			
Data Hold Time (Notes 8, 9)	t _{HO,DAT}	Fast mode	0		0.9	μs
		Standard mode	0		0.9	
Data Setup Time (Note 10)	t _{SU,DAT}	Fast mode	100			ns
		Standard mode	250			
START Setup Time	t _{SU,STA}	Fast mode	0.6			μs
		Standard mode	4.7			
Rise Time of Both SDA and SCL Signals (Note 11)	t _R	Fast mode	20 +		300	ns
		Standard mode	0.1C _B		1000	
Fall Time of Both SDA and SCL Signals (Note 11)	t _F	Fast mode	20 +		300	ns
		Standard mode	0.1C _B		300	
Setup Time for STOP Condition	t _{SU,STO}	Fast mode	0.6			μs
		Standard mode	4.7			
Capacitive Load for Each Bus Line	C _L	(Note 11)			400	pF
Capacitance for SDA, SCL	C _{I/O}			10		pF
Pulse Width of Spikes That Must Be Suppressed by the Input Filter	t _{SP}			30		ns
Pushbutton Debounce	PB _{DB}			250		ms
Reset Active Time	t _{RAST}			250		ms
Oscillator Stop Flag (OSF) Delay	t _{OSF}	(Note 12)		100		ms
Temperature Conversion Time	t _{CONV}			125	200	ms

Power-Switch Characteristics

(T_A = T_{MIN} to T_{MAX})

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V _{CC} Fall Time: V _{PP(MAX)} to V _{PP(MIN)}	t _{VCCF}		300			μs
V _{CC} Rise Time: V _{PP(MIN)} to V _{PP(MAX)}	t _{VCCR}		0			μs
Recovery at Power-Up	t _{REC}	(Note 13)		250	300	ms

Block Diagram



Pin Description

PIN	NAME	FUNCTION
1	32kHz	32kHz Output. This open-drain pin requires an external pullup resistor. When enabled, the output operates on either power supply. It may be left open if not used.
2	V _{CC}	DC Power Pin for Primary Power Supply. This pin should be decoupled using a 0.1µF to 1.0µF capacitor. If not used, connect to ground.
3	INT/SQW	Active-Low Interrupt or Square-Wave Output. This open-drain pin requires an external pullup resistor connected to a supply at 5.5V or less. This multifunction pin is determined by the state of the INTCON bit in the Control Register (0Eh). When INTCON is set to logic 0, this pin outputs a square wave and its frequency is determined by R32 and R81 bits. When INTCON is set to logic 1, then a match between the timekeeping registers and either of the alarm registers activates the INT/SQW pin (if the alarm is enabled). Because the INTCON bit is set to logic 1 when power is first applied, the pin defaults to an interrupt output with alarms disabled. The pullup voltage can be up to 5.5V, regardless of the voltage on V _{CC} . If not used, this pin can be left unconnected.
4	RST	Active-Low Reset. This pin is an open-drain input/output. It indicates the status of V _{CC} relative to the V _{DD} specification. As V _{CC} falls below V _{DD} , the RST pin is driven low. When V _{CC} exceeds V _{DD} for t _{RES} , the RST pin is pulled high by the internal pullup resistor. The active-low, open-drain output is combined with a debounced pushbutton input function. This pin can be activated by a pushbutton reset request. It has an internal 50kΩ nominal value pullup resistor to V _{CC} . No external pullup resistors should be connected. If the oscillator is disabled, t _{RES} is bypassed and RST immediately goes high.
5-12	N.C.	No Connection. Must be connected to ground.
13	GND	Ground
14	V _{BAT}	Backup Power-Supply Input. When using the device with the V _{BAT} input as the primary power source, this pin should be decoupled using a 0.1µF to 1.0µF low-leakage capacitor. When using the device with the V _{BAT} input as the backup power source, the capacitor is not required. If V _{BAT} is not used, connect to ground. The device is UL recognized to ensure against reverse charging when used with a primary lithium battery. Go to www.maximintegrated.com/qsi/info/ul .
15	SDA	Serial Data Input/Output. This pin is the data input/output for the I2C serial interface. This open-drain pin requires an external pullup resistor. The pullup voltage can be up to 5.5V, regardless of the voltage on V _{CC} .
16	SCL	Serial Clock Input. This pin is the clock input for the I2C serial interface and is used to synchronize data movement on the serial interface. Up to 5.5V can be used for this pin, regardless of the voltage on V _{CC} .

Detailed Description

The DS3231 is a serial RTC driven by a temperature-compensated 32kHz crystal oscillator. The TCXO provides a stable and accurate reference clock, and maintains the RTC to within ±2 minutes per year accuracy from -40°C to +85°C. The TCXO frequency output is available at the 32kHz pin. The RTC is a low-power clock/calendar with two programmable time-of-day alarms and a programmable square-wave output. The INT/SQW provides either an interrupt signal due to alarm conditions or a square-wave output. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap

year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. The internal registers are accessible through an I2C bus interface.

A temperature-compensated voltage reference and comparator circuit monitors the level of V_{CC} to detect power failures and to automatically switch to the backup supply when necessary. The RST pin provides an external pushbutton function and acts as an indicator of a power-fail event.

Operation

The block diagram shows the main elements of the DS3231. The eight blocks can be grouped into four functional groups: TCXO, power control, pushbutton function, and RTC. Their operations are described separately in the following sections.

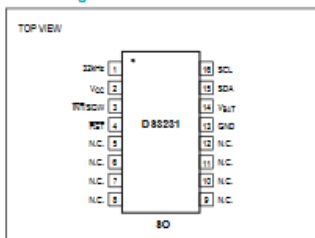
Handling, PCB Layout, and Assembly

The DS3231 package contains a quartz tuning-fork crystal. Pick-and-place equipment can be used, but precautions should be taken to ensure that excessive shocks are avoided. Ultrasonic cleaning should be avoided to prevent damage to the crystal.

Avoid running signal traces under the package, unless a ground plane is placed between the package and the

signal line. All N.C. (no connect) pins must be connected to ground.

Moisture-sensitive packages are shipped from the factory dry packed. Handling instructions listed on the package label must be followed to prevent damage during reflow. Refer to the IPC/JEDEC J-STD-020 standard for moisture-sensitive device (MSD) classifications and reflow profiles. Exposure to reflow is limited to 2 times maximum.

Pin Configuration**Chip Information**

SUBSTRATE CONNECTED TO GROUND
PROCESS: CMOS

Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
DS3231B#	0°C to +70°C	16 SO
DS3231BN#	-40°C to +85°C	16 SO

#Denotes an RoHS-compliant device that may include lead (Pb) that is exempt under RoHS requirements. The lead finish is JESD97 category e3, and is compatible with both lead-based and lead-free soldering processes. A 'B' anywhere on the part mark denotes an RoHS-compliant device.

Package Information

For the latest package outline information and land patterns (footprints), go to www.maximintegrated.com/packages. Note that a "u", "H", or "P" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

PACKAGE TYPE	PACKAGE CODE	OUTLINE NO.	LAND PATTERN NO.
16 SO	W16SH2	21-0042	80-0107

B.2. DATASHEET SIM900

1

 **Itead Studio**
Make Innovation easier

Tech Support: support@iteadstudio.com

IComSat v1.1

-SIM900 GSM/GPRS shield

Overview



IComsat is a GSM/GPRS shield for Arduino and based on the SIM900 Quad-band GSM/GPRS module. It is controlled via AT commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands), and fully compatible with Arduino / Iteaduno and Mega.

Features

- Quad-Band 850/900/1800/1900MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
- Class 4 (2W@850/900MHz)
- Class 1 (1W@1800/1900MHz)
- Control via commands (GSM 07.07, 07.05 and SIMCOM enhanced AT Commands)
- Short message service
- Free serial port selection
- All SIM900 pins breakout
- RTC supported with Super Cap
- Power on/off and reset function supported by Arduino interface

IComSat V1.0

iteadstudio.com

05.27.2011

Specifications

PCB size	77.2mm X 66.0mm X 1.6mm
Indicators	PWR, status LED, net status LED
Power supply	5~20V, compatible with Arduino
Communication Protocol	UART
RoHS	Yes

Electrical Characteristics

Specification	Min	Type	Max	Unit
Power Voltage (Vlogic)	4.5	5	5.5	VDC
Power Voltage (Vsupply)	9	-	20	VDC
Input Voltage VH:	4.5	5	5.5	V
Input Voltage VL:	-0.3	0	0.5	V
Current Consumption (pulse)	-	-	2000	mA
Current Consumption (Continuous)			500	mA
Baud rate		9600		bps

Hardware

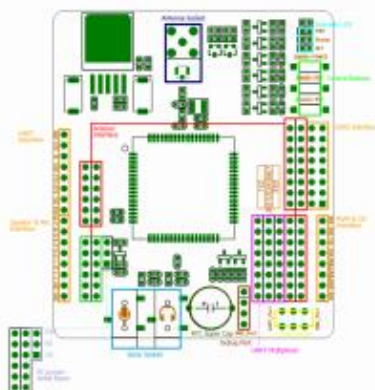


Figure 1 Top Map

Interface	Pin	Description
GPIOs	1	VDD*
	2	GND
	3	GPIO1
	4	GPIO2
	5	GPIO3
	6	GPIO4
	7	GPIO5
	8	GPIO6
	9	GPIO7
	10	GPIO8
	11	GPIO9
	12	GPIO10
	13	GPIO11
	14	GPIO12

	15	GND
	16	VDD*
UART	1	GND
	2	DTR
	3	RI
	4	DCD
	5	CTS
	6	RTS
	7	TXD
	8	RXD
IIC&PWM	1	GND
	2	GND
	3	IIC_SCL
	4	IIC_SDA
	5	PWM2
	6	PWM1
	7	GND
	8	GND
Debug_Port	1	GND
	2	PERKEY
	3	DBG_RXD
	4	DBG_TXD

VDD* = 3.0V

Note 1: the operation level of the port is 3.0V

Installation

UART Multiplexer (For free UART connection setting)

You can use the jumper to connect the TXD and RXD pins on SIM to any pins of Arduino D0 - D7.



Figure 3 UART Multiplexer

When using the connection as Figure 4, the SIM900 connect to the ATmega328 chip on board.



Figure 4 Connect the Arduino board

When you using the connection as Figure 5, the SIM900 connect with the FT232RL chip, and the FT232RL connect to PC by USB. With this configuration you can use the serial software on PC to control or configure the SIM900 module.



Figure 5 Connect the UART Interface as FT232

Except the 2 configurations above, you can connect the TXD and RXD to any other pins from D0-D7, and using the software-serial library to control the SIM900 module.



Figure 6 an example for software-serial connection

Com Switch

There is a com switch for selecting the communicate port to UART port or Debug port of SIM900 module as the following figure 7. When connect to the SIM900 debug port, the UART multiplexer just can be set as Figure 4 or 5.



Figure 7-a



Figure 7-b

RI Behavior Output

There is a Jumper for configuration of RI, one to connect D2, another to connect D14.

Status	RI response
Standby	High
Voice call	The pin is changed to low. When any of the following events occur, the pin will be changed to high: (1) Establish the call (2) Hang up the call
Data call	The pin is changed to low. When any of the following events occur, the pin will be changed to high: (1) Establish the call (4) Hang up the call
SMS	The pin is changed to low, and kept low for 120ms when a SMS is received. Then it is changed to high.
URC	The pin is changed to low, and kept low for 120ms when some URCs are reported. Then it is changed to high.

The behavior of the RI pin is shown in the following figure when SIM900 module is used as a receiver.

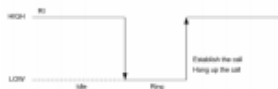


Figure 8 RI behavior of voice calling as a receiver

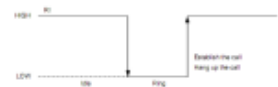


Figure 9 RI behaviors of data calling as a receiver



Figure 10 RI behavior of URC or receive SMS

When the IComSat is used as caller, the RI will remain high.

Indicator LED and Buttons:

NETSTATUS: The status of the NETSTATUS LED is listed in following table:

Status	Description
Off	SIM900 is not running
64ms On/800ms Off	SIM900 not registered the network
64ms On/3000ms Off	SIM900 registered to the network
64ms On/300ms Off	GPRS communication is established

STATUS: Power status of SIM900.

PWR: Power status of IcomSat.

SIM900-POWER: After the IcomSat power on, you need to press the SIM900-POWER button for a moment to power on the SIM900 module. The pin 9 of Arduino interface is connecting to PWRKEY, and a high pulse with 400us wide can power on/off it.

SIM900-RST: Reset the SIM900. The pin 10 of Arduino interface is connecting to RESRT of SIM900, and a high pulse with 400us wide can power on/off it.

Arduino-RST: Reset the Arduino.

Revision History

Rev.	Description	Release date
v1.0	Initial version	2011-5-27
v1.1	Add the R1 description, and fix some typo	2011-7-22
v1.2	Update for IComsat v1.1	2012-1-17

B.3. DATASHEET Arduino Mega

MEGA 2560 R3 Datasheet



■ INTRODUCTION

MEGA 2560 R3 is an Arduino compatible board, it's based on Arduino MEGA 2560 design. So you can use BUONO MEGA 2560 as Arduino MEGA 2560. All Code, shield and IDE for Arduino MEGA 2560 R3 are also valid on BUONO MEGA 2560. Some visible improvement on hardware make BUONO more flexible and easier use. For example: 3.3V or 5V IO selectable allow you connect some 3.3V modules (such as XBee) to BUONO directly.



■ THE DIFFERENCE BETWEEN Arduino MEGA 2560 AND BUONO MEGA 2560

	Arduino MEGA 2560	BUONO MEGA 2560
USB Connector	Type B Female	Micro USB connector
Operating Voltage	5V	3.3V or 5V selectable
3.3V Current	50mA	600mA
5V Current	500mA	1.8A
Input Range	7V to 12V	6V to 23V
MCU	ATMEGA2560	ATMEGA2560
Others		Ground terminal LPF for AVCC

■ FEATURES

- Inherits all of Arduino MEGA 2560's features
- Compatible to Arduino MEGA 2560's pin layout, screw hole and dimensions
- 3.3V or 5V operating voltage selectable
- More visible location of indication LEDs
- Evolved with SMD components
- Digital I/O pins 54 (of which 15 provided PWM output)
- Analog Input pins 16
- Flash memory 256 KB or which 8 KB used by bootloader
- SRAM 8 KB
- EEPROM 4 KB
- Clock speed 16 MHz
- Micro USB connection make the cable can be mixed use with most digital products such as cell phone
- 5V DC/DC provide 95% efficiency
- 6V to 23V wide range input
- Provide max 1.8A for peripheral circuits
- Ground terminal make measurement easier

■ PACKAGE LIST

- 1x BUONO MEGA 2560
- 1x Micro USB cable
- 4x Screw and nut

■ CONTACT US

540 Mill River LN San Jose,
CA 95134
E-mail: support@inhaos.com
Web: www.inhaos.com

B.4 DATASHEET Sensor Arus YHDC SCT-013

Split-Core Current Transformer

Model: SCT-013 Series

Characteristics: Opening size: 13mm×13mm

Non-linearity: ±3% (10%—120% rated input current)

1.5m leading wire, Φ3.5 three core plug standard output

Current output type or Voltage output type (Voltage output type built-in sampling resistor)

Purpose: Suitable for the current measuring
monitoring and protection of AC motor
lighting equipment air compressor and so on

Core material: Ferrite

Mechanical strength: The number of switching is not less than 1000 times
(Test under 20°C)

Safety index: Dielectric strength (between Shell and output) 6000V AC/1min

Fire resistance property: in accordance with UL94-V0

Working temperature: -25°C—+70°C

Outline size diagram: (in mm)

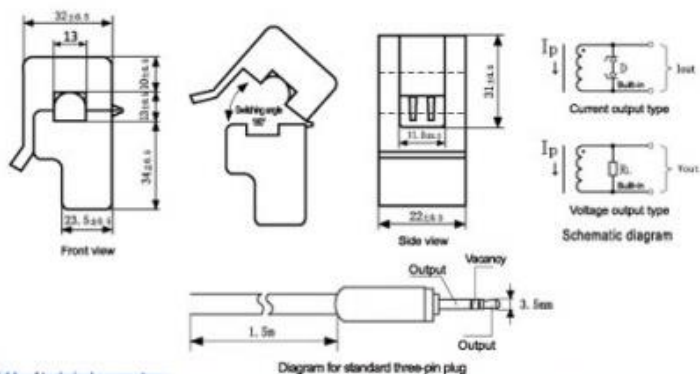


Table of technical parameters:

Model	SCT-013-000	SCT-013-005	SCT-013-010	SCT-013-015	SCT-013-020
Input current	0-100A	0-5A	0-10A	0-15A	0-20A
Output mode	Current/33m A	Voltage/1V	Voltage/1V	Voltage/1V	Voltage/1V
Model	SCT-013-025	SCT-013-030	SCT-013-050	SCT-013-060	SCT-013-070
Input current	0-25A	0-30A	0-50A	0-60A	
Output mode	Voltage/1V	Voltage/1V	Voltage/1V	Voltage/1V	

Output mode: Voltage output type built-in sampling resistor; Current output type built-in protective diode;
Forbidden to be opening operating for current type.

DAFTAR RIWAYAT HIDUP



Nama : Krisna Sadewa
TTL : Ponorogo, 10 Agustus 1996
Jenis Kelamin : Laki - Laki
Agama : Islam
Alamat : Ds. Sidoharjo, Kec. Pulung Kab. Ponorogo
Telp/HP : 085607551160
E-mail : kriznasadewo@gmail.com

RIWAYAT PENDIDIKAN

1. 2002 – 2008 : MI Muhammadiyah 7
2. 2008 – 2011 : SMP Negeri 2 Ponorogo
3. 2011 – 2014 : SMA Negeri 1 Ponorogo
4. 2014 – 2017 : D3 Teknik Elektro Otomasi, Program Studi Teknik Elektro Komputer Kontrol – Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS)

PENGALAMAN KERJA

1. Kerja Praktek di PT PLN (Persero) Rayon Dukuh Kupak Area Surabaya Selatan

PENGALAMAN ORGANISASI

1. Staff Divisi Minat dan Bakat HIMAD3TEKTRO 2015-2016

-----*Halaman ini sengaja dikosongkan*-----

DAFTAR RIWAYAT HIDUP



Nama : Iqbal Asad Kutsuro
TTL : Madiun, 07 Januari 1996
Jenis Kelamin : Laki-Laki
Agama : Islam
Alamat : JL. Panji No.50 Nglames,
Kab. Madiun
Telp/HP : 081330087465
E-mail : askutoo@gmail.com

RIWAYAT PENDIDIKAN

1. 2002 – 2008 : MI Islamiyah 01 Madiun
2. 2008 – 2011 : SMP Negeri 3 Madiun
3. 2011 – 2014 : SMA Negeri 2 Madiun
4. 2014 – 2017 : D3 Teknik Elektro Otomasi, Program Studi Teknik Listrik PLN – Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS)

PENGALAMAN KERJA

1. Kerja Praktek di PT PLN (Persero) APD Kota Madiun
2. Kerja Praktek di PT PLN (Persero) Rayon Dukuh Kupang

PENGALAMAN ORGANISASI

1. Staff Departemen Komunikasi dan Informasi HIMAD3TEKTRO 2015-2016
2. Sie Akomodasi IARC 2015
3. Sie Akomodasi IARC 2016
4. Ketua Organisasi Eksternal *Mobile Legend sns squad* 2017
5. Staff Komunikasi dan Informasi Forum Mahasiswa Madiun Surabaya 2016
6. Penanggung Jawab Eksternal FORMAD *Goes To School* 2016

-----*Halaman ini sengaja dikosongkan*-----